Общие сведение о качестве обслуживания (QoS) для коммутаторов серии Catalyst 6000

Содержание

Введение

В документе объясняются возможность качества обслуживания (QoS), доступные в коммутаторах семейства Catalyst 6000. В данном документе описываются возможности настройки QoS и приводится несколько примеров реализации QoS.

Этот документ не предназначен, чтобы быть руководством по конфигурации. Примеры конфигурации используются всюду по этой бумаге для помощи в пояснении Характеристик QoS программного и аппаратного обеспечения Семейства Catalyst 6000. Для ссылки синтаксиса для структур команды QoS, см. следующую конфигурацию и руководства по командам для Семейства Catalyst 6000:

- Коммутаторы семейства Catalyst 6500

Определение слоя 2 QoS

Многие думают, что QoS на коммутационном уровне 2 (L2) всего лишь обслуживает приоритеты кадров Ethernet, но не все понимают, что за этим кроется нечто большее. L2 QoS влечет за собой придерживающееся:

1. **Планирование Входной очереди:** когда кадр приходит в порт, он поступает в одну из очередей порта для дальнейшей коммутации в выходной порт. Как правило, множественные очереди используются, где другой трафик требует других уровней сервиса, или где задержка коммутатора должна быть сведена к минимуму. Например, IP базировал видео, и голосовые данные требуют низкой задержки,

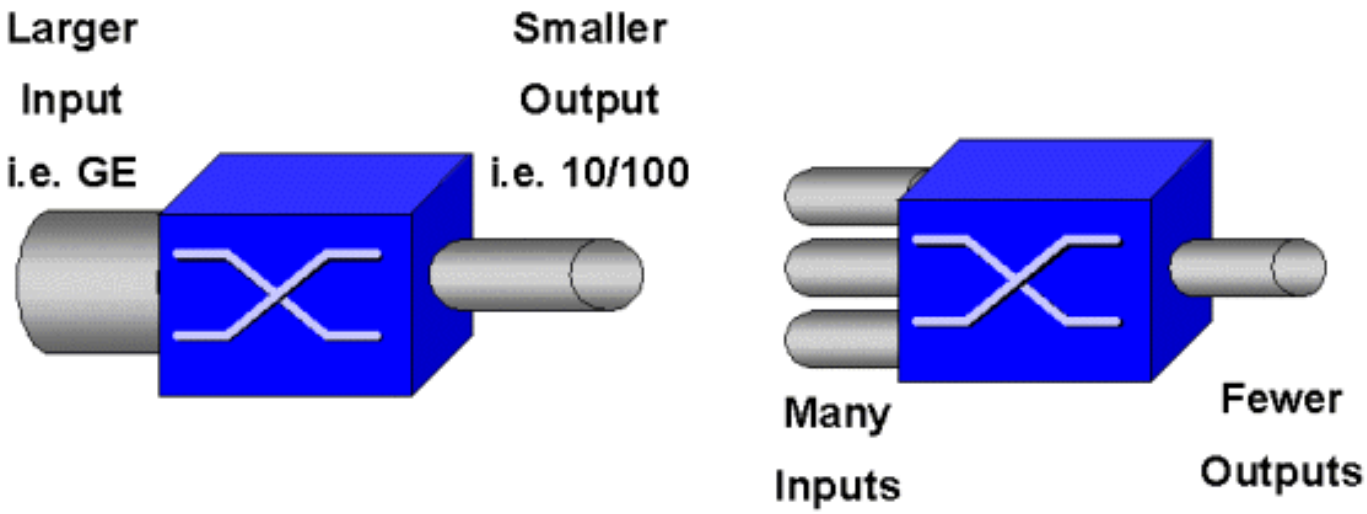
таким образом, может быть потребность коммутировать эти данные до коммутации других данных, таких как Протокол FTP, сеть, электронная почта, Telnet, и так далее.

2. **Классификация:** процесс классификации вовлекает осматривающие другие поля в заголовок Ethernet L2, наряду с полями в IP - заголовке (Уровень 3 (L3)) и Протокол управления передачей / Протокол датаграммы пользователя (TCP/UDP) заголовок (Уровень 4 (L4)) для помощи в определении уровня обслуживания, который будет применен к кадру, поскольку это передает транзитом коммутатор.
3. **Применение политик:** применение политик является процессом осмотра Фрейма Ethernet, чтобы видеть, превысило ли это предварительно определенную скорость трафика в течение определенного выделенного интервала времени (как правило, на этот раз структурируйте, фиксированный номер, внутренний к коммутатору). Если тот кадр внепрофилен (т.е. это - часть потока данных сверх предварительно определенного предела скорости), это может быть или отброшено или значение класса обслуживания (CoS), может быть отмечен.
4. **Перезапись:** процесс перезаписи дает возможность коммутатору изменять CoS в заголовке Ethernet или биты типа обслуживания (ToS) в заголовке IPV4.
5. **Планирование выходной очереди:** после процессов перезаписи коммутатор помещает кадр Ethernet в соответствующую исходящую очередь (на выходе) для коммутации. Коммутатор будет выполнять управление буфером в этом запросе, чтобы убедиться, что буфер не переполнен. Это будет, как правило, делать это путем использования алгоритма Random Early Discard (RED), посредством чего случайные кадры удалены (отброшенные) из очереди. Взвешенный RED (WRED) – это модификация RED (используемая некоторыми модулями в семействе Catalyst 6000), в соответствии с ним значения CoS просматриваются, чтобы определить, какие кадры будут отброшены. Когда буферы достигают предварительно заданных порогов, более низкоприоритетные кадры обычно отбрасываются, а более высокоприоритетные кадры остаются в очереди.

Этот документ объясняет более подробно каждый из механизмов выше и как они касаются Семейства Catalyst 6000 в следующих разделах.

Необходимость QoS в коммутаторе

Огромные объединительные платы, миллионы коммутируемых пакетов в секунду и неблокировочных переключателей все синонимичны со многими коммутаторами сегодня. Какая польза от QoS? Ответ: из-за перегрузки.



Коммутатор может быть самым быстрым коммутатором в мире, но если у вас будет любой из этих двух сценариев, показанных на рисунке выше, то тот коммутатор испытает перегрузку. Во времена перегрузки, если средства управления перегрузками сети не существуют, будут отброшены пакеты. Если пакеты отброшены, то происходит повторная передача. При возникновении повторной передачи сетевая загрузка может расти. В сетях, которые уже переполнены, это может добавить к существующим проблемам производительности и потенциально далее ухудшить производительность.

Для сходящихся сетей управление перегрузками становится еще важнее. Если задержки понесены, на чувствительный к задержкам трафик задержки, такой как голос и видео можно сильно повлиять. Просто добавление большего количества буферов к коммутатору также не обязательно облегчит проблемы перегрузки. Чувствительный к задержкам трафик задержки должен быть коммутирован максимально быстро. Во-первых, необходимо определить этот важный трафик через методы классификации, и затем внедрить способы управления буферами для предотвращения трафика более высокого приоритета от того, чтобы быть отброшенным во время перегрузки. Наконец, необходимо включить способы планирования для коммутации важных пакетов от очередей как можно быстрее. Поскольку вы будете читать в этом документе, Семейство Catalyst 6000 внедряет все эти способы, делая его подсистему QoS одним из самых всесторонних в отрасли сегодня.

Все методы QoS, описанные в предыдущем разделе, будут исследоваться более подробно всюду по этому документу.

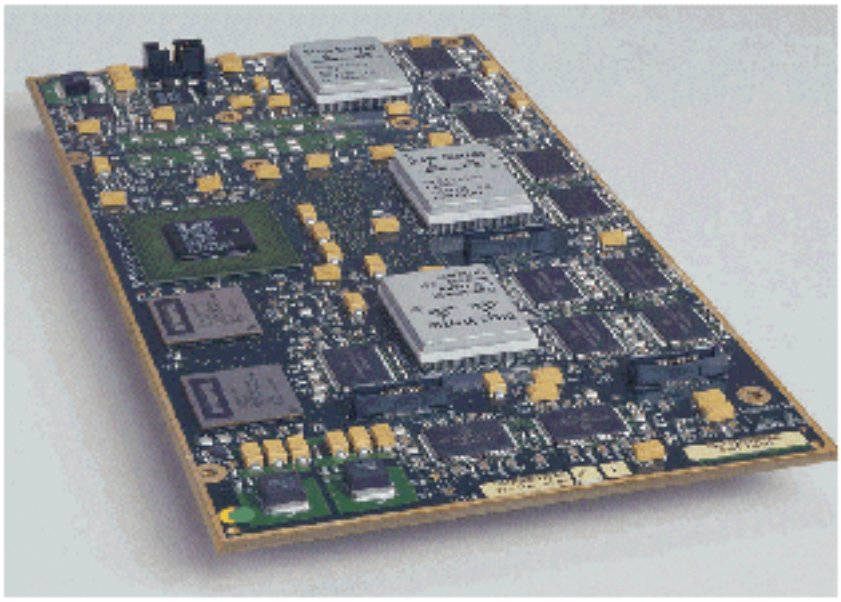
Аппаратная поддержка QoS в семействе устройств Catalyst 6000

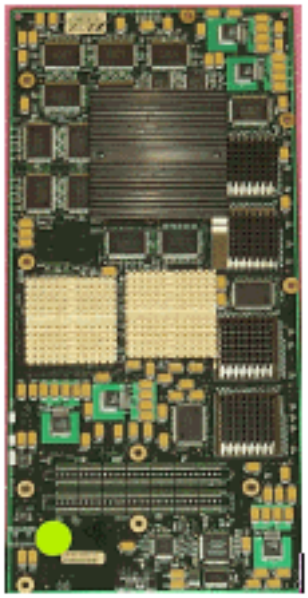
Для поддержки QoS в Семействе Catalyst 6000 некоторая аппаратная поддержка требуется. Аппаратные средства, которые поддерживают QoS, включают Функциональную Карту Многоуровневого Коммутатора (MSFC), Policy Feature Card (PFC) и Приложение для порта Определенные Интегральные схемы (ASIC-схемы) на самих линейных картах. В этом документе не рассматриваются возможности QoS MSFC, а содержатся данные о возможностях QoS PFC и ASIC на линейных картах.

PFC

PFC версии 1 представляет собой дочернюю плату на основе Supervisor I (SupI) и Supervisor

IA (SupIA) серии Catalyst 6000. Плата PFC2 – это модернизированная версия платы PFC1, она поставляется с новым модулем Supervisor II (SupII) и некоторыми новыми встроенными ASIC. В то время как и PFC1 и PFC2 прежде всего известны их аппаратным ускорением коммутации L3, QoS является одной из их других целей. PFC показывают ниже.





Не смотря на то, что PFC 1 и PFC2 по сути эквивалентны, они имеют различия в функциональности QoS. А именно, PFC2 добавляет придерживающееся:

1. Возможность сместить политику QoS на плату распределенной переадресации (DFC).

2. Решения по политикам слегка отличаются. И PFC1 и PFC2 поддерживают обычное применение политик, посредством чего кадры отброшены или отмечены, если агрегат или политика микропотока возвращают внепрофильное решение. Однако PFC2 добавляет поддержку превышения скорости, которое указывает на второй уровень применения политик, в котором могут быть взяты действия политики.

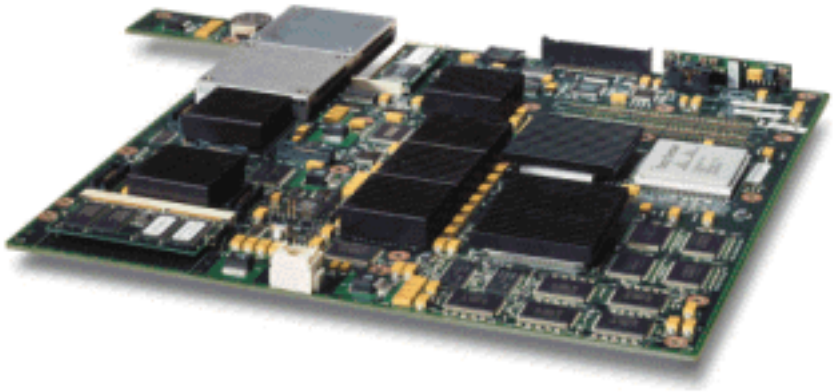
Когда ограничитель превышения скорости определен, пакеты могут быть отброшены или отмечены, когда они превышают превышение скорости. Если уровень избыточного контроля установлен, избыточное Сопоставление DSCP используется для замены исходного DSCP-

значения вниз отмеченным значением. Если только обычный уровень политики установлен, обычное Сопоставление DSCP используется. Когда оба уровня политики будут установлены, уровень избыточного контроля будет иметь приоритеты для выбора правил сопоставления.

Следует отметить, что функции QoS, описанные в этом документе, выполненном ASIC-схемами, упомянули высокий уровень урожая производительности. Производительность QoS в базовой семье Catalyst 6000 (без модуля матрицы коммутации) возвращает значение 15 MPPS. Если DFC используются, дополнительный прирост производительности может быть достигнут для QoS.

DFC

Например, можно подключить DFC к WS-X6516-GBIC. Однако это - стандартный прибор на карте WS-X6816-GBIC. Это может также поддерживаться на других будущих оптоволоконных линейных картах, таких как недавно представленная матрица 10/100 (WS-X6548-RJ45) линейная карта, оптоволоконная линейная карта RJ21 (WS-X6548-RJ21), и 100FX линейная карта (WS-X6524-MM-FX). Плата DFC изображена ниже.



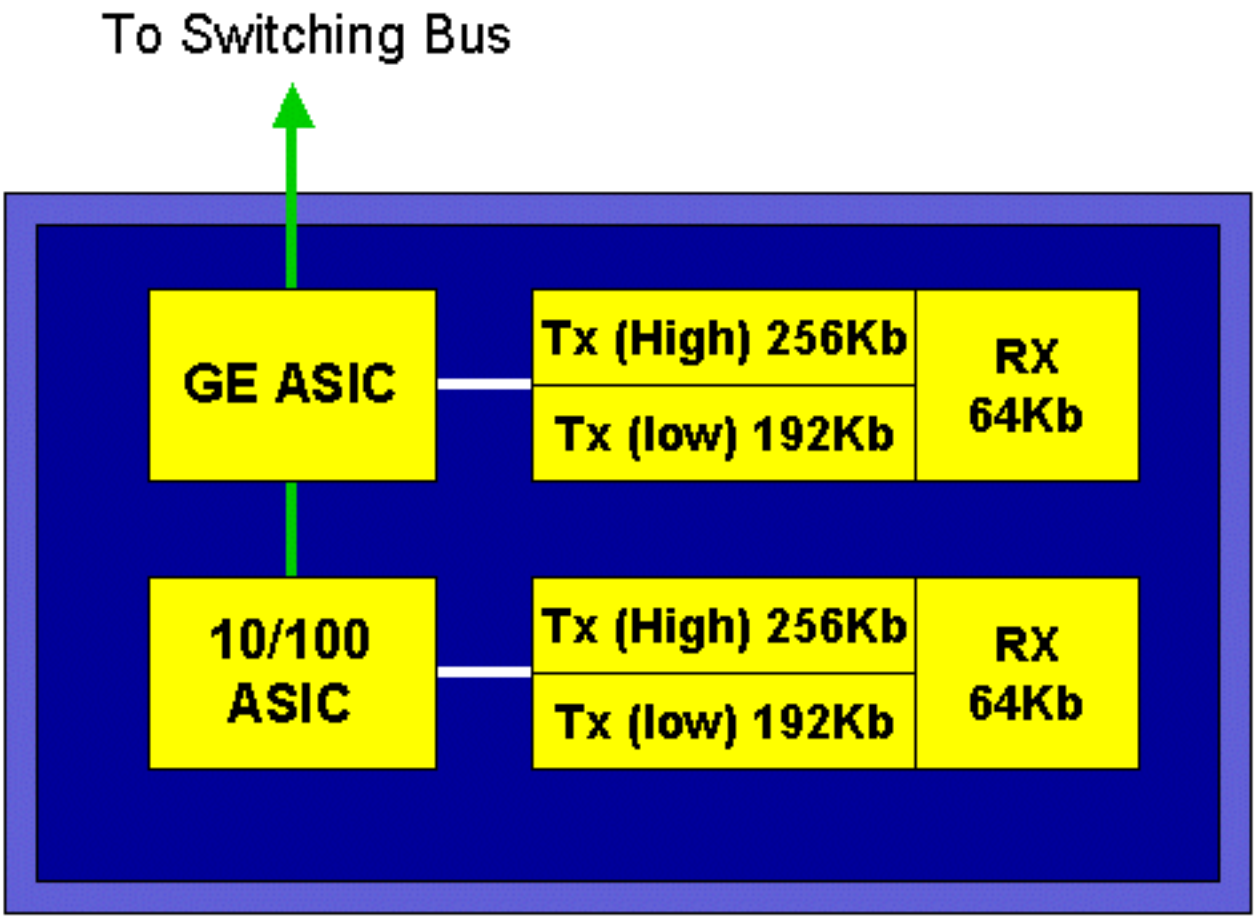
DFC позволяет матрице (связанная перемычка) линейная карта выполнять локальный коммутатор. Чтобы сделать это, это должно также поддержать любые политики QoS, которые были определены для коммутатора. Администратор не может непосредственно настроить DFC; скорее это прибывает под контролем основного MSFC/PFC на активном управляющем модуль. Основной PFC оттолкнет таблицу Базы данных переадресации (FIB), которая дает DFC ее L2 и таблицы пересылки L3. Это также оттолкнет копию политик QoS так, чтобы они были также локальны для линейной карты. Последующий за этим, решения локального коммутатора могут сослаться на локальную копию любых политик QoS, предоставляющих аппаратные скорости обработки QoS и приводящих к более высоким уровням производительности хотя распределенная коммутация.

Порт базирующиеся ASIC-схемы

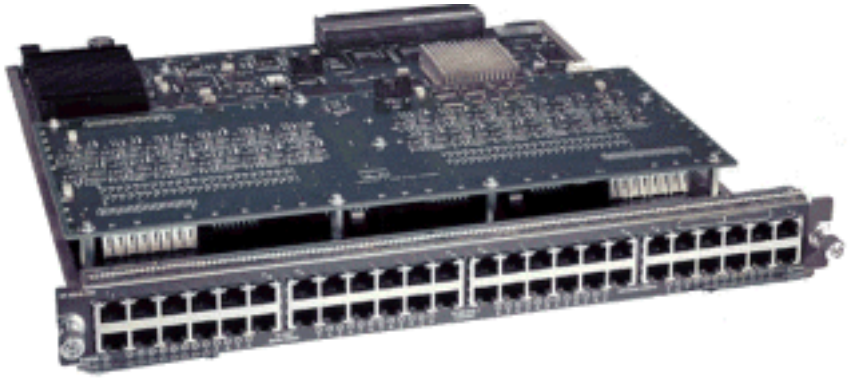
Чтобы закончить аппаратную картину, каждая линейная карта реализует несколько ASIC. Данные ASIC выполняют функции системы обслуживания очередей, буферизации и пороговых значений, которые используются для временного хранения кадров при передаче через коммутатор. На платах 10/100 комбинация различных ASIC используется для обеспечения 48 портов 10/100.

Исходные 10/100 Линейные карты (WS-X6348-RJ45)

ASIC 10/100 предоставляют серии очередей приема (Rx) и передачи (TX) для каждого порта 10/100. ASICs предоставляет 128 Кб буферизации на каждые 10/100 портов. См. Комментарии к выпуску для подробных данных о том, что на буферизацию порта доступно на каждой линейной карте. Каждый порт на этой линейной карте поддерживает одну очередь Rx, и две очереди TX обозначили высокий и низкий. Это показывают в приведенном ниже рисунке.



В схеме выше, каждый 10/100 ASIC предоставляет выход для 12 10/100 портов. Для каждого 10/100 порта предоставлены 128 буферов K. 128 K буферов разделены между каждой из этих трех очередей. Цифры, показанные в приведенной выше очереди, не являются значениями по умолчанию, а скорее представлением возможной конфигурации. Единый Rx очередь занимает 16K и остаточная память (112K) разделяется между двумя Tx очередями. По умолчанию (в CatOS), высокая очередь получает 20 процентов этого пространства, и низкая очередь получает 80 процентов. В Catalyst IOS по умолчанию должен дать высокой очереди 10 процентов и низкой очереди 90 процентов.

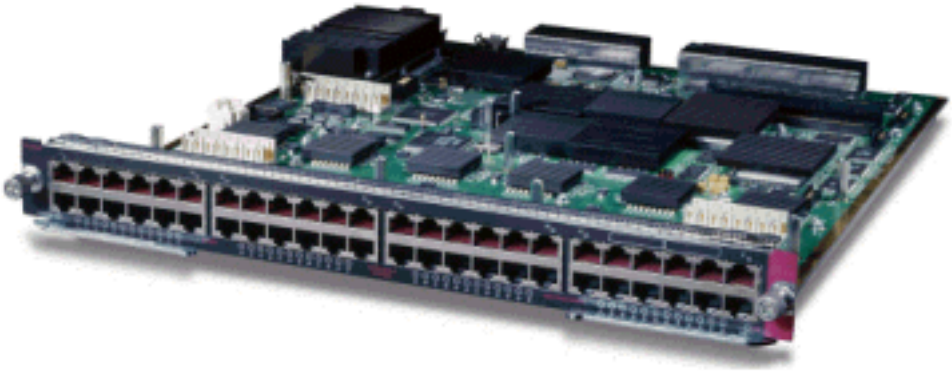


В то время как карта предоставляет двухступенчатую буферизацию, только 10/100 ASIC

базировался, буферизация доступна, чтобы быть манипулированной во время конфигурации QoS.

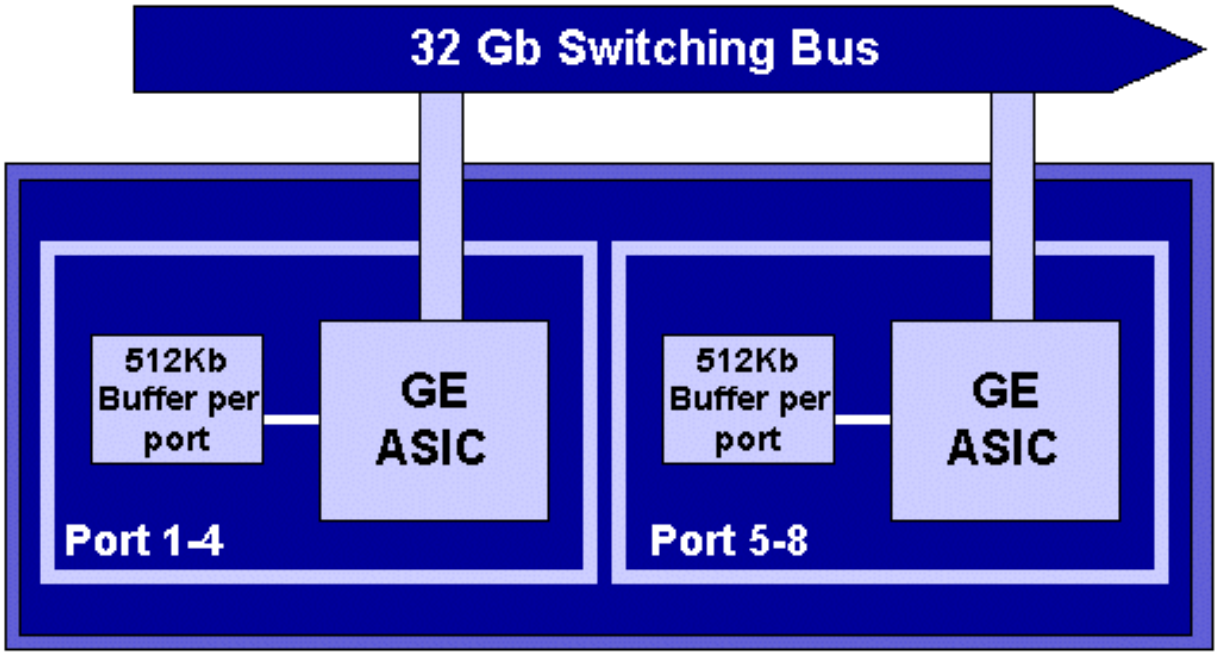
Матрица 10/100 Линейные карты (WS-X6548-RJ45)

Новые ASIC 10/100 предоставляют серии очередей на прием и на передачу для каждого порта 10/100. ASIC-схемы предоставляют совместно используемый пул памяти, доступной через 10/100 порты. См. Комментарии к выпуску для подробных данных о том, что на буферизацию порта доступно на каждой линейной карте. Каждый порт на этой линейной карте поддерживает две очереди Rx и три очереди TX. Одна очередь Rx и одна очередь TX каждый обозначены как очередь абсолютного приоритета. Она используется как очередь с низкой задержкой, которая идеально подходит для трафика, чувствительного к задержке, например VoIP.

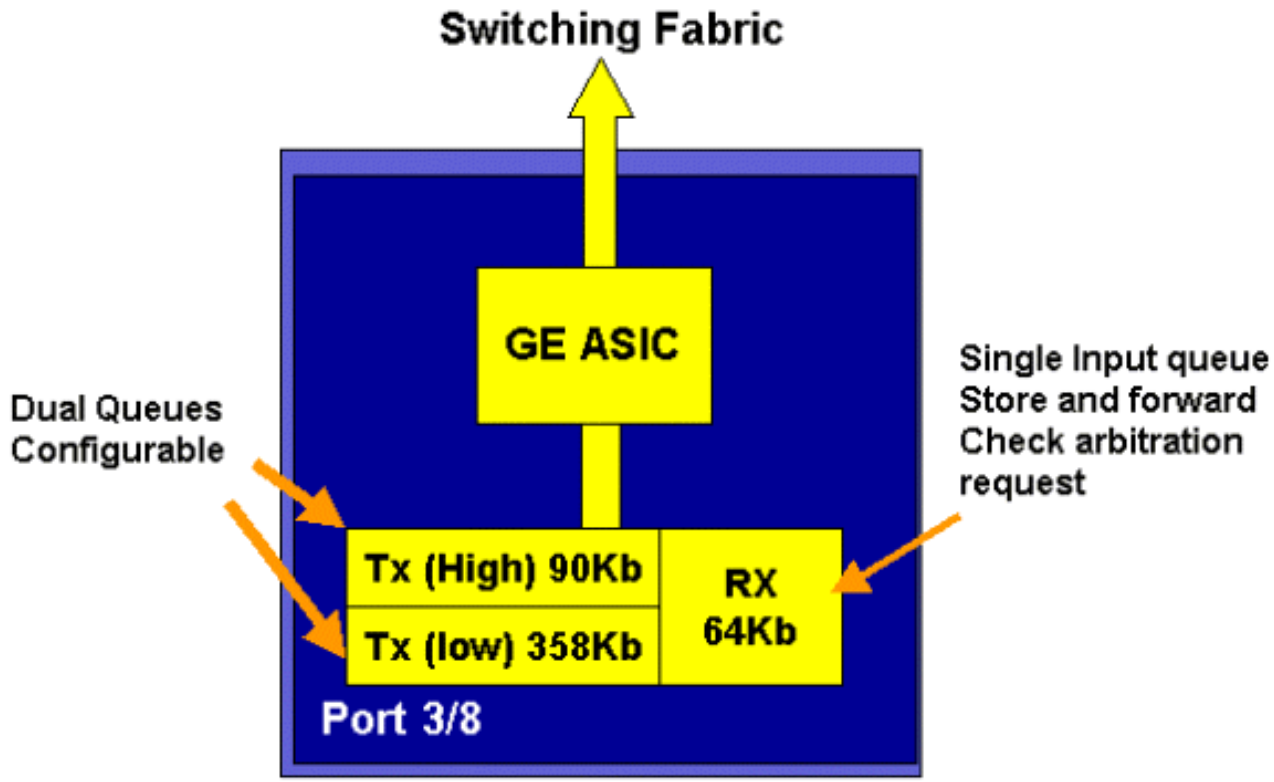


Линейные карты GE (WS-X6408A, WS-X6516, WS-X6816)

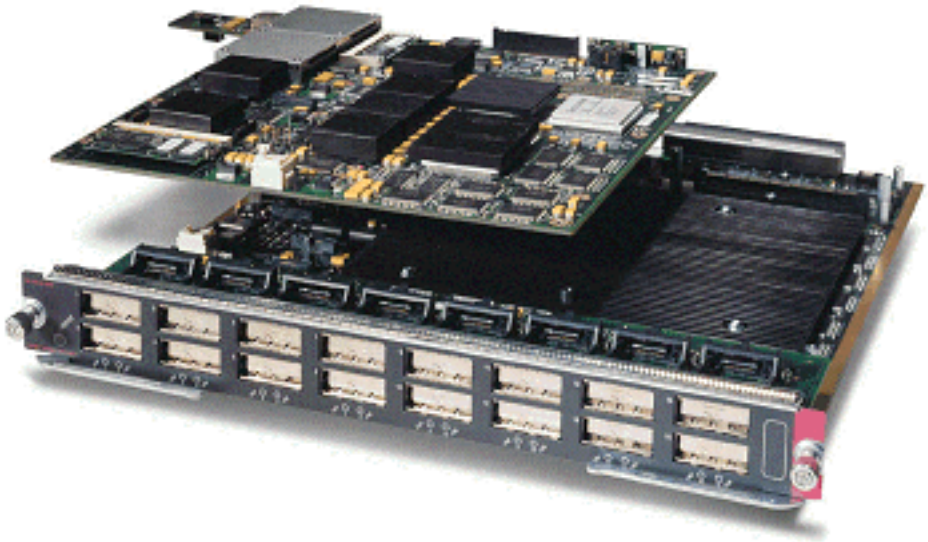
Для линейных карт GE ASIC предоставляет 512 K на буферизацию порта. Представление восьмипортовой линейной карты GE показывают в приведенном ниже рисунке.



Как с 10/100 портами, каждый порт GE имеет три очереди, один Rx и две очереди TX. Для линейной платы WS-X6408-GBIC это конфигурация по умолчанию, которая показана на приведенной ниже схеме.



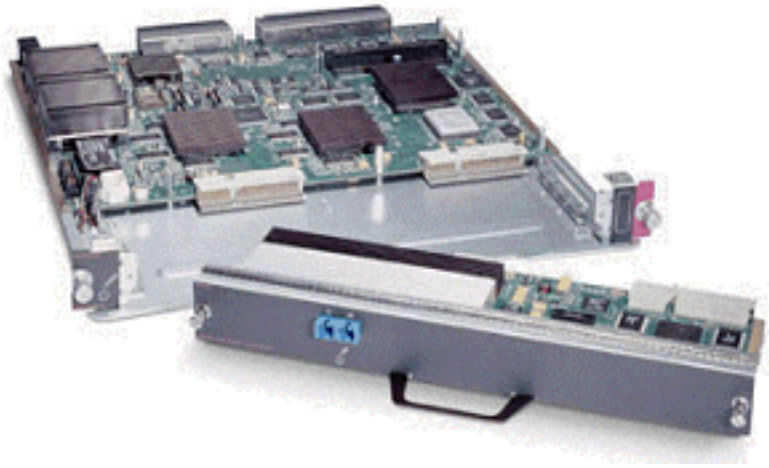
Для новых плат GE с 16 портами, портов GBIC модулей SupIA и SupII, а также платы GE WS-X6408A-GBIC с 8 портами предусмотрено две дополнительных очереди с жестким приоритетом (SP). Одна очередь SP назначается в качестве очереди приема (Rx), а другая – в качестве очереди передачи (TX). Эта очередь SP используется прежде всего для организации очереди чувствительного к задержкам трафика задержки, такого как голос. При использовании очереди SP любые данные, помещенные в эту очередь, будут обработаны перед данными в старшей и младшей очередях. Только то, когда очередь SP пуста, будет высокие и низкие очереди обслуживаться.



10 линейных карт GE (WS-X6502-10GE)

В последней половине 2001 Cisco представила ряд 10 линейных карт GE, предоставляющих один порт 10 GE на линейную карту. Этот модуль берет один слот от 6000 шасси. 10 линейных карт GE поддерживают QoS. Для 10 портов GE это предоставляет две очереди

Rx и три очереди TX. Одна очередь Rx и одна очередь TX каждый назначены как очередь SP. Буферизация также предоставлена для порта, предоставив в общей сложности 256 K Буфера RX и 64 МБ Буфера передачи. Данный порт реализует структуру очереди 1p1q8t для стороны Rx и структуру очереди 1p2q1t для стороны TX. Структуры очередей будут рассмотрены более подробно далее в этой документации.



Сводка аппаратного обеспечения QoS семейства Catalyst 6000

Аппаратные компоненты, которые выполняют вышеупомянутые функции QoS в Семействе Catalyst 6000, детализированы в таблице ниже.

QoS Process	Catalyst 6500 Component that performs function
Input Scheduling	Performed by port ASIC's L2 only with or without the PFC
Classification	Performed by Supervisor or PFC L2 only is done by Supervisor L2/3 is done by PFC
Policing	Done by PFC via L3 forwarding Engine
Packet Re-write	Done by port ASIC's L2/L3 based on classification done in point 2 above
Output Scheduling	Done by port ASIC's L2/L3 based on classification done in point 2 above

Программная поддержка системы QoS коммутаторами серии Catalyst 6000

Семейство Catalyst 6000 поддерживает две операционных системы. Исходная платформа программного обеспечения CatOS была разработана на основе базы кодов, используемой на платформе Catalyst 5000. Позже, Cisco представила Интегрированный Cisco IOS ® (Режим работы в собственной системе команд) (ранее известный как Native IOS), который использует основание кода, полученное из IOS маршрутизатора Cisco. Обе Платформы операционной системы (CatOS и Интегрированный Cisco IOS (Режим работы в собственной системе команд)) внедряют поддержку программного обеспечения для включения QoS на платформе семейства Коммутатора Catalyst 6000 с помощью аппаратных средств,

описанных в предыдущих разделах.

Примечание: В этом документе используются примеры конфигурации для обеих платформ ОС.

Механизмы приоритетов в IP и Ethernet

Для любых сервисов QoS, которые будут применены к данным, должен быть способ пометить или расположить по приоритетам пакет IP или Фрейм Ethernet. ToS и поля CoS используются для достижения этого.

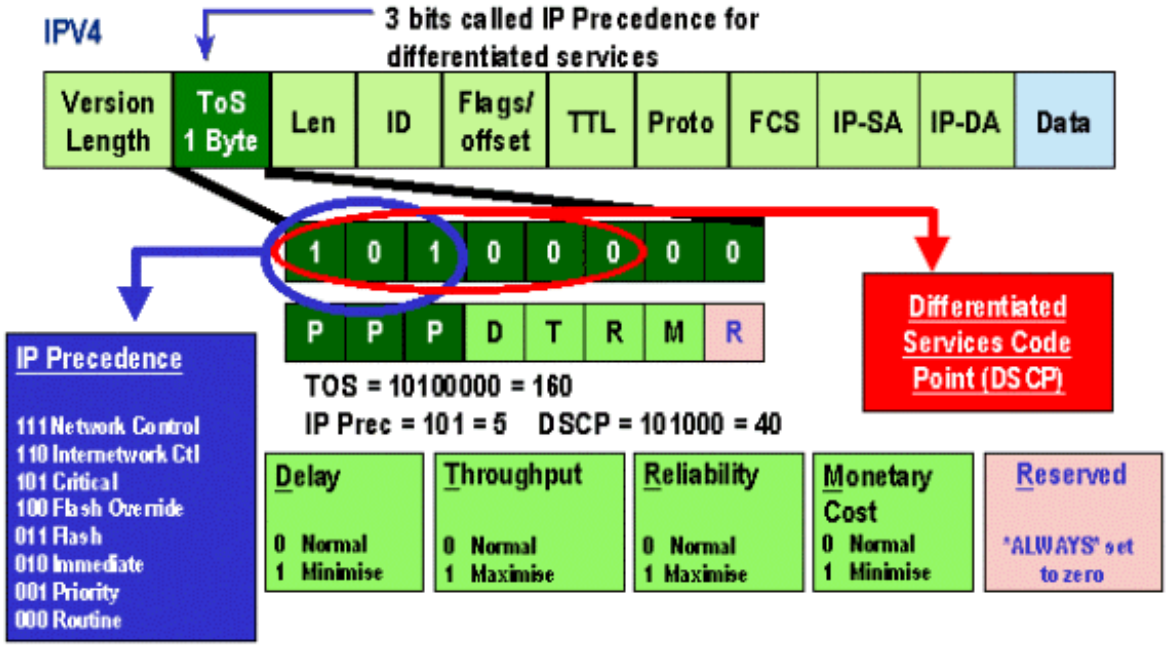
ToS

ToS является однобайтовым полем, которое существует в заголовке IPv4. Поле ToS состоит из восьми битов, первый три из которых используются для указания приоритета IP пакета. Эти первые три разряда называются битами приоритета IP-адреса. Эти биты могут быть установлены от нуля до семь с нулем, являющимся самый низким приоритетом и семь являющийся наивысшим приоритетом. В течение многих лет доступна поддержка для настройки приоритета IP в IOS. Поддержка сброса приоритета IP-адресов может быть выполнена с помощью MSFC или PFC (независимо от MSFC). Параметр доверия недоверяемых может также вытереть любые Параметры приоритета IP на входящем фрейме.

Для приоритета IP можно задать следующие значения:

IP Precedence bits	IP Precedence Value
000	Routine
001	Priority
010	Intermediate
011	Flash
100	Flash Override
101	Critical
110	Internetwork Control
111	Network Control

На приведенной ниже схеме показаны двоичные значения приоритета IP-адреса в заголовке ToS. Три наиболее значимых бита (MSB) интерпретируются как биты предшествования IP.



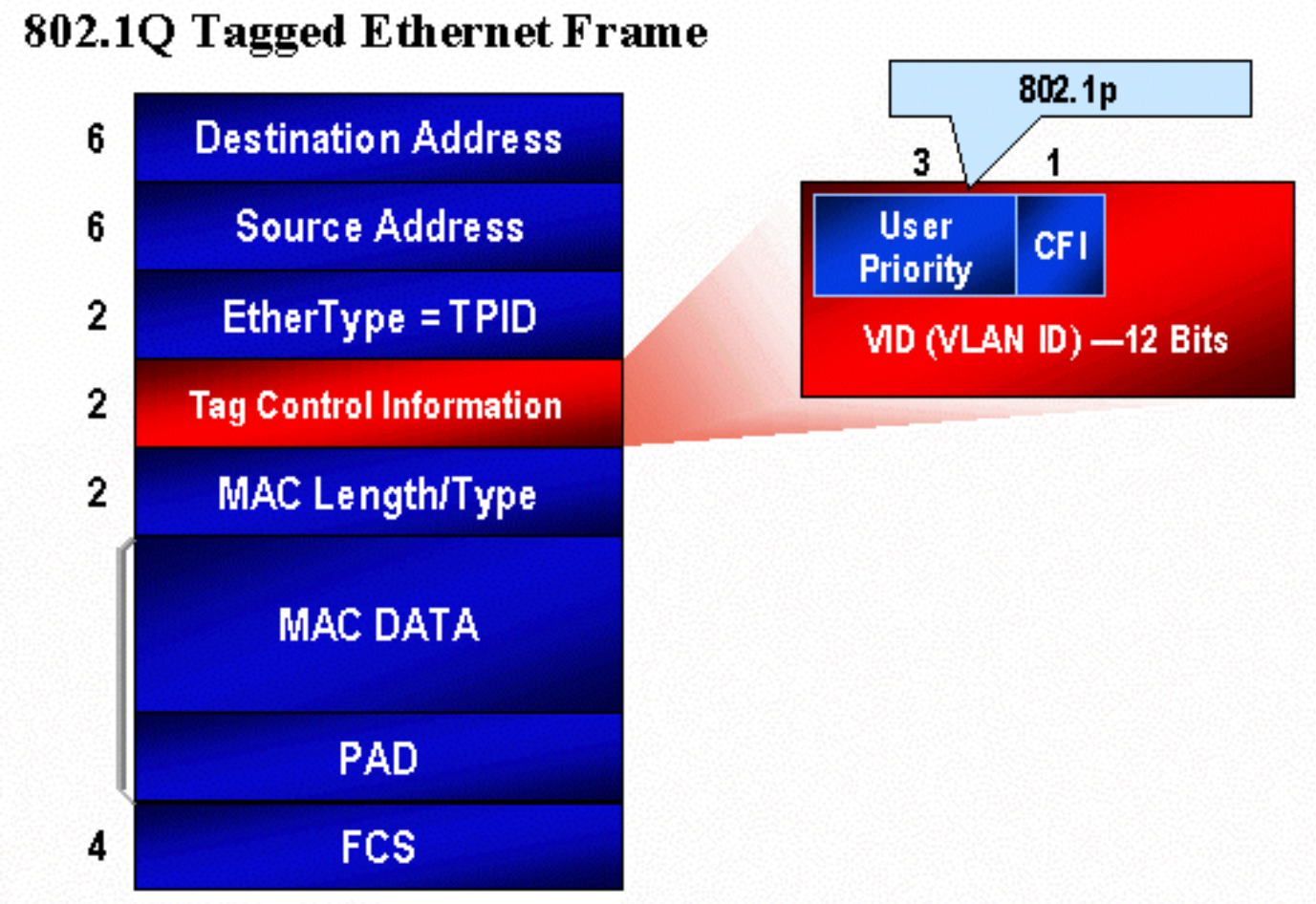
Позже, использование поля ToS было расширено для затрагивания этих шести MSB, называемых DSCP. DSCP приводит к 64 значениям приоритета (два к питанию шесть), который может быть назначен на пакет IP.

Семейство Catalyst 6000 может управлять ToS. Этого можно достичь, используя PFC и/или MSFC. Когда кадр приходит в коммутатор, ему назначается значение DSCP. Данное значение DSCP используется внутри коммутатора для назначения уровней обслуживания (политики QoS), заданных администратором. DSCP уже может существовать в кадре и использоваться, или DSCP может быть получена из существующего CoS, приоритета IP либо DSCP в кадре (порт должен быть надежным). Карта используется внутренне в коммутаторе для получения DSCP. С восемью возможными приоритетами CoS/IP и 64 возможными значениями DSCP, карта по умолчанию будет сопоставлять CoS/IPPrec 0 с DSCP 0, CoS/IPPrec 1 с DSCP 7, CoS/IPPrec 2 с DSCP 15 и т.д. Эти сопоставления по умолчанию могут быть отвергнуты администратором. Если кадр должен быть отослан на порт исходящих соединений, класс обслуживания CoS может быть переписан, а для получения нового CoS может быть использовано значение DSCP.

CoS

CoS обращается к трем битам или в заголовке ISL или в заголовке 802.1Q, которые используются для указания на приоритет Фрейма Ethernet, поскольку это проходит через коммутируемую сеть. В целях этого документа мы только обращаемся к использованию заголовка 802.1Q. Биты CoS в заголовке 802.1Q обычно называются биты 802.1p. Не удивительно, существует три бита CoS, который совпадает с количеством битов, используемых для приоритета IP-трафика. Во многих сетях, для поддержания QoS End to End пакет может пересечь и L2 и домены L3. Для поддержки QoS ToS может быть сопоставлен с CoS, а CoS может быть сопоставлен с ToS.

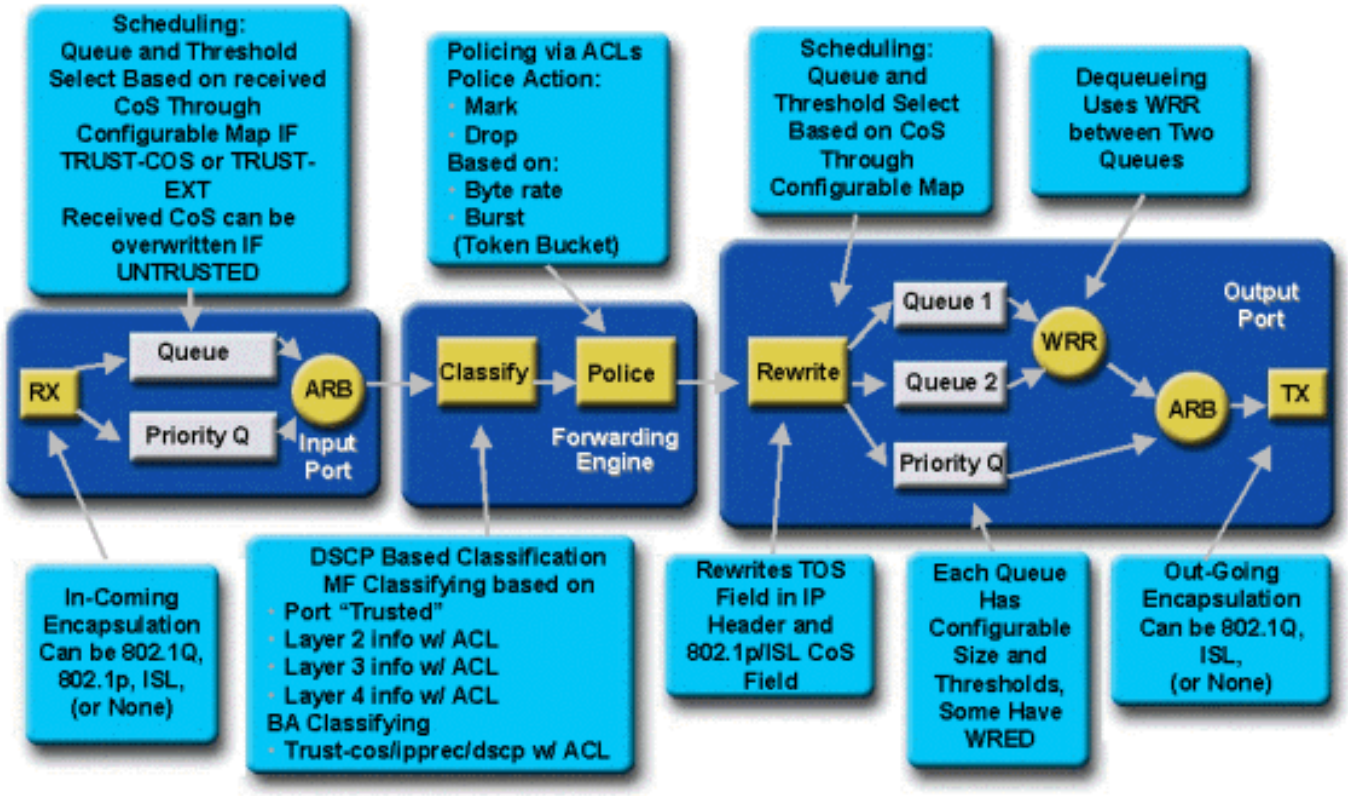
Приведенная ниже схема является кадром Ethernet с добавленным полем 802.1Q, состоящим из двухбайтового Ethertype и двухбайтового тега. В двухбайтовой метке биты приоритета пользователя (известный как 802.1p).



Поток QoS в коммутаторах семейства Catalyst 6000

QoS в семействе Catalyst 6000 представляет собой наиболее полную реализацию QoS из всех текущих коммутаторов Cisco Catalyst. Следующие разделы описывают, как различные процессы QoS применены к кадру, поскольку он передает транзитом коммутатор.

Ранее в этом документе, было обращено внимание, что существует много элементов QoS, которые могут предложить много L2 и коммутаторы L3. К этим элементам относятся классификация, планирование входящей очереди, применение политик, перезапись и планирование исходящей очереди. Отличие от семейства Catalyst 6000 состоит в том, что эти элементы QoS применяются механизмом L2, который может понимать данные L3 и L4, как данные заголовка L2. Следующая схема иллюстрирует, как эти элементы реализуются в семействе Catalyst 6000.



Кадр вводит коммутатор и первоначально обработан ASIC порта, который принял кадр. Это разместит кадр в очередь Rx. В зависимости от линейной карты Семейства Catalyst 6000 будут одна или две очереди Rx.

Порт ASIC использует CoS биты как индикатор очереди для кадра (при множественных входящих очередях). Если порт классифицирован как недоверяемый, ASIC порта может перезаписать существующие биты CoS на основе предварительно определенного значения.

Затем кадр передается механизму передачи L2/L3 (PFC) для классификации и контроля (ограничения скорости). Классификация является процессом присвоения кадра DSCP-значение, которое используется внутренне коммутатором для обработки кадра. DSCP будет получен из одного из придерживающегося:

1. Существующее значение DSCP, установленное до входа фрейма в коммутатор
2. Принятые биты предшествования IP уже установлены в заголовке IPv4. Как существует 64 DSCP-значения и только восемь значений приоритета IP-трафика, администратор настроит сопоставление, которое используется коммутатором для получения DSCP. Если администратор не настроит отображения, действуют отображения по умолчанию.
3. Полученным данным уже установлено CoS до поступления кадра на коммутатор. Подобно приоритетам IP-адреса, максимальное число значений CoS равно восьми, каждое из них должно соответствовать одному из 64 значений DSCP. Можно настроить это сопоставление или использовать уже имеющееся сопоставление по умолчанию.
4. Настройте кадр при помощи значения DSCP по умолчанию, обычно присвоенного через запись списка управления доступом (ACL).

После того, как DSCP-значение назначено на кадр, применение политик (ограничение скорости) применено, должен настройка ограничителя скорости существовать. Применение политики ограничит поток данных через PFC благодаря отбрасыванию или маркировке

трафика, не соответствующего профилю. Внепрофильный термин, использованный, чтобы указать, что трафик превысил предел, определенный администратором как сумма битов в секунду, которые передаст PFC. Можно отбросить внепрофильный трафик или понизить значение CoS. PFC1 и PFC2 поддерживают только ограничение входящего трафика (ограничение скорости). Поддержка политик по входящим и исходящим пакетам будет доступна в новом выпуске PFC.

PFC тогда передаст кадр к выходному порту для обработки. На этом этапе процесс перезаписи вызван для изменения значений CoS в кадре и значения ToS в заголовке IPv4. Это получено из внутреннего DSCP. Затем кадр будет помещён в очередь передачи на основании его значения CoS, готовым к передаче. Пока кадр находится в очереди, порт ASIC отслеживает буферы и выполняет WRED, чтобы избежать переполнения буферов. Алгоритм планирования WRR тогда используется, чтобы планировать и передать кадры от выходного порта

Каждый из разделов ниже исследует этот поток, более подробно дающий примеры конфигурации для каждого из шагов, описанных выше.

Очереди, буферы, пороги и сопоставления

Прежде чем конфигурация QoS описана подробно, определенные сроки должны быть объяснены далее, чтобы гарантировать, что вы полностью понимаете возможности конфигурации QoS коммутатора.

Очереди

Каждый порт на коммутаторе имеет серию очередей ввод/вывода, которые используются в качестве областей временного хранения для данных. Линейные карты семейства Catalyst 6000 внедряют другое количество очередей для каждого порта. Очереди обычно реализуются в аппаратных ASIC для каждого порта. На картах каналов из первого поколения семейства Catalyst 6000 типичная конфигурация состояла из одной входящей очереди и двух исходящих. На более новых линейных картах (10/100 и GE), ASIC внедряет дополнительный набор двух очередей (один ввод и выходные данные) приводящий к двум входным очередям и трем очередям вывода. Эти две дополнительные очереди являются особыми SP-очередями, которые используются для обработки чувствительного к задержкам трафика – например, в системе VoIP. Они обслуживаются в режиме SP. Если кадр поступает в очередь SP, планирование обработки кадров из очередей с более низким приоритетом прекращается для выполнения обработки кадра в очереди SP. Планирование пакетов в более низких очередях возобновляется только после опустошения очереди SP.

Если фрейм прибывает на порт (для ввода или вывода) во время перегрузки, он будет помещен в очередь. Решение, на основании которого кадр помещается в последовательность, обычно производится на основании значения CoS в заголовке Ethernet входящего кадра.

На исходящем порту для опустошения очереди TX (вывод) будет применен алгоритм планирования. WRR является техникой, используемой для достижения этого. Для каждой очереди надбавка используется для диктовки, сколько данные будут освобождены от очереди прежде, чем перейти на следующую очередь. Вес, назначенный администратором - это число от 1 до 255, назначенное каждой очереди TX.

Буферы

Каждой очереди назначают определенная величина пространства буфера для хранения транзитных данных. Резидент на порте ASIC – это память, которая выделяется и распределяется по портам. Для каждого порта GE ASIC GE назначает 512 K пространства буфера. Для 10/100 портов ASIC порта резервирует 64 K или 128 K (в зависимости от линейной карты) на буферизацию порта. Это пространство буфера затем делится между очередью Rx (входной) и очередью TX (выходной).

Пороги

Единственный аспект нормальной передачи данных – удаление пакета приводит к тому, что пакет пересылается заново (движение TCP). Во времена перегрузки это может добавить к нагрузке сети и потенциально заставить буферы перегружаться еще больше. Поскольку средство обеспечения, что буферы не переполняются, Коммутатор семейства Catalyst 6000 Family, использует много способов для предотвращения этого от случая.

Пороги являются воображаемыми уровнями, назначенными коммутатором (или администратор), которые определяют точки использования, в которых алгоритм управления перегрузками сети может начать отбрасывать данные от очереди. На портах семейства Catalyst 6000 обычно существуют четыре порога, которые связаны с входными очередями. Обычно существует два порога, привязанные к очередям вывода.

Данные пороги развертываются также в контексте QoS как способ назначения этим порогам кадров с различными приоритетами. Поскольку буфер начинает заполняться, и пороги нарушены, администратор может сопоставить другие приоритеты с другими порогами, указывающими к коммутатору, какие кадры должны быть отброшены, когда превышен порог.

Сопоставления

В очередях и пороговые разделы выше, было упомянуто, что значение CoS во Фрейме Ethernet используется для определения, какая очередь разместить кадр в и в, какая точка заполнения буфера является кадром, имеющим право быть отброшенным. Это назначение сопоставления.

При настройке QoS в коммутаторах семейства Catalyst 6000 активизируются стандартные отображения, которые определяют следующие параметры:

- при каких порогах кадры с указанными значениями CoS могут быть отброшены
- то, которые помещают кадр в очередь, размещено в (на основе его значения CoS)

Пока существуют сопоставления по умолчанию, они могут быть заменены администратором. Сопоставление существует для придерживающегося:

- Значения CoS на входящем фрейме к DSCP-значению
- Значения приоритета IP-трафика на входящем фрейме к DSCP-значению
- DSCP-значения к значению CoS для исходящего кадра
- Значения CoS в пороги отбрасывания на очередях приема
- Значения CoS в пороги отбрасывания на очередях передачи
- Снижение приоритета DSCP оценивает для кадров, которые превышают операторы применения политик
- Значения CoS к кадру с определенным MAC - адресом назначения

WRED и WRR

WRED и WRR – два особо продуктивных алгоритма семейства Catalyst 6000. И WRED и WRR используют тег приоритета (CoS) во Фрейме Ethernet для обеспечения улучшенного управления буферами и планирования исходящего трафика. В

WRED

WRED является алгоритмом управления буферами, используемым Семейством Catalyst 6000 для уменьшения влияния понижающегося трафика с высоким приоритетом во времена перегрузки. WRED основывается на Алгоритме RED.

Для понимания КРАСНОГО и WRED, пересмотрите понятие управления потока TCP. Управление потоками данных гарантирует, что отправитель TCP не сокрушает сеть. Алгоритм медленного пуска TCP является частью решения. Это диктует, что, когда поток запускается, один пакет передается, прежде чем это будет ждать подтверждения. До получения ACK отправлены два пакета. Таким образом, постепенно возрастает число пакетов, отправленных до получения каждого ACK. Это продолжится, пока поток не достигает уровня передачи (т.е. передает *x* количество пакетов), который сеть может обработать без перегрузки несения загрузки. Если перегрузка произойдет, то алгоритм медленного пуска возвратит размер окна к исходному состоянию (т.е. количество пакетов, переданных прежде, чем ждать подтверждения), таким образом уменьшая общую производительность для того сеанса TCP (поток).

RED будет наблюдать за очередью по мере ее наполнения. Как только определенный порог был превышен, пакеты начнут отбрасываться случайным образом. Никакое отношение не дано определенным потокам; скорее случайные пакеты будут отброшены. Эти пакеты могут быть из потоков с высоким или низким приоритетом. Отброшенные пакеты могут быть частью единого потока или множественных потоков TCP. Если на множественные потоки влияют, как описано выше, это может оказать значительное влияние на каждый размер окна потоков.

В отличие от алгоритма RED, WRED не столь случаен при отбрасывании кадров. WRED учитывает приоритет кадров (в случае с семейством Catalyst 6000 он использует значение CoS). При использовании WRED администратор устанавливает для кадров с определенными значениями QoS соответствующие пороги. После превышения данных порогов назначенные им кадры со значениями CoS можно отбросить. Другие кадры со значениями CoS, назначенными на более высокие пороговые значения, остаются в очереди. Этот процесс обеспечивает потоки более высокого приоритета, которые будут сохранены неповрежденным хранением их больших неповрежденных размеров окна и уменьшение задержки, вовлеченной в получение пакетов с отправителя на получатель.

Как вы знаете, поддерживает ли ваша линейная карта WRED? Выполните следующую команду. В выходных данных проверьте для раздела, который указывает на поддержку WRED на том порту.

```
Console> show qos info config 2/1 QoS setting in NVRAM: QoS is enabled Port 2/1 has 2 transmit queue with 2 drop thresholds (2q2t). Port 2/1 has 1 receive queue with 4 drop thresholds (1q4t). Interface type:vlan-based ACL attached: The qos trust type is set to untrusted. Default CoS = 0 Queue and Threshold Mapping: Queue Threshold CoS ----- --------- -------------- 1 1 0 1 1 2 2 3 2 1 4 5 2 2 6 7 Rx drop thresholds: Rx drop thresholds are disabled for untrusted ports. Queue # Thresholds - percentage (abs values) ------- ---------------------------------- 1 50% 60% 80%
```

```
100% TX drop thresholds: Queue # Thresholds - percentage (abs values) ------- ------------------
------------------- 1 40% 100% 2 40% 100% TX WRED thresholds: WRED feature is not supported for
this port_type. !-- Look for this. Queue Sizes: Queue # Sizes - percentage (abs values) -------
----------------------------------- 1 80% 2 20% WRR Configuration of ports with speed
1000MBPS: Queue # Ratios (abs values) ------- ----------------------------------- 1 100 2 255
Console> (enable)
```

Если WRED не доступен на порту, порт будет использовать метод отбрасывания остатка управления буферами. Отбрасывание остатка, подразумевается его имя, просто сбрасывает входящие кадры, когда буферы полностью использованы.

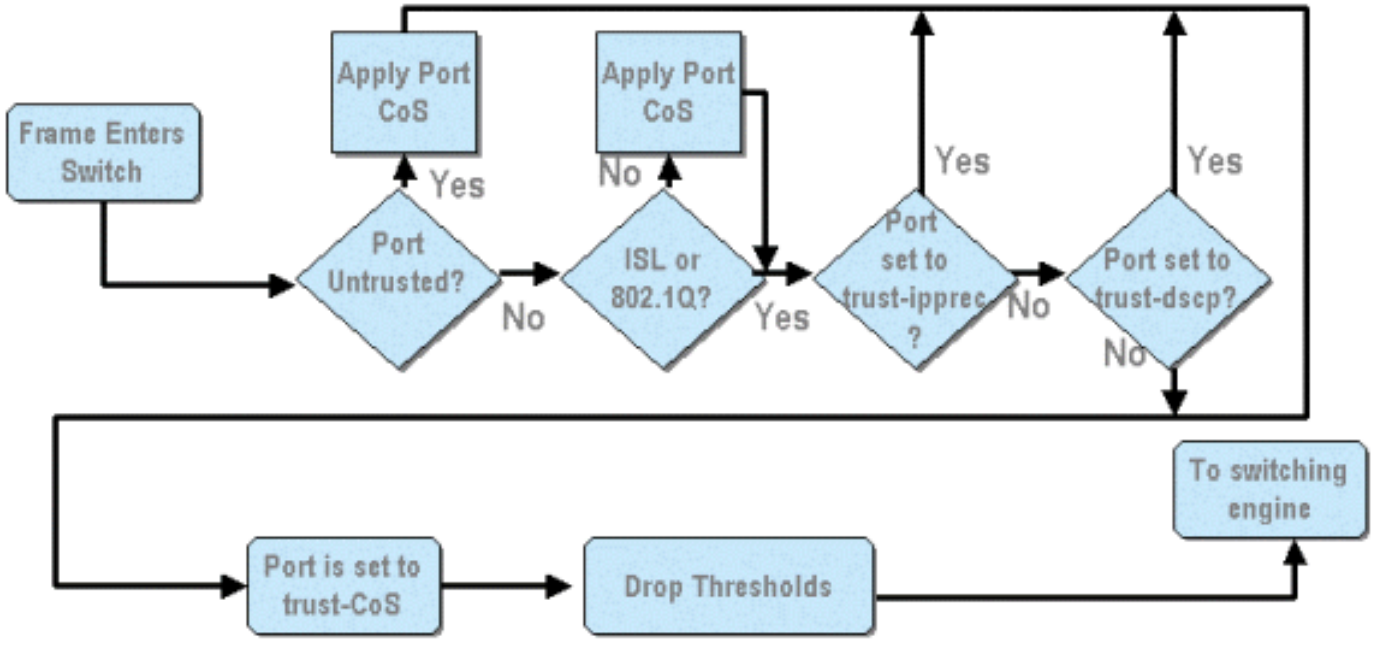
WRR

WRR используется для планирования выходного трафика от очередей TX. Обычный кольцевой алгоритм Round robin чередуется между очередями TX, передающими равное число пакетов от каждой очереди прежде, чем переместиться к следующей очереди. Взвешенный аспект WRR позволяет алгоритму планирования проверять взвешивание, присвоенное очереди. Благодаря этому определенные очереди могут получать доступ к полосе пропускания большего размера. Алгоритм планирования WRR опустошит больше данных от определенных очередей, чем другие очереди, таким образом предоставляя уклон для назначенных очередей.

Конфигурация для WRR и других аспектов того, что было описано выше, объяснена в следующих разделах.

Настройка QoS для специализированной ИС порта для семейства Catalyst 6000

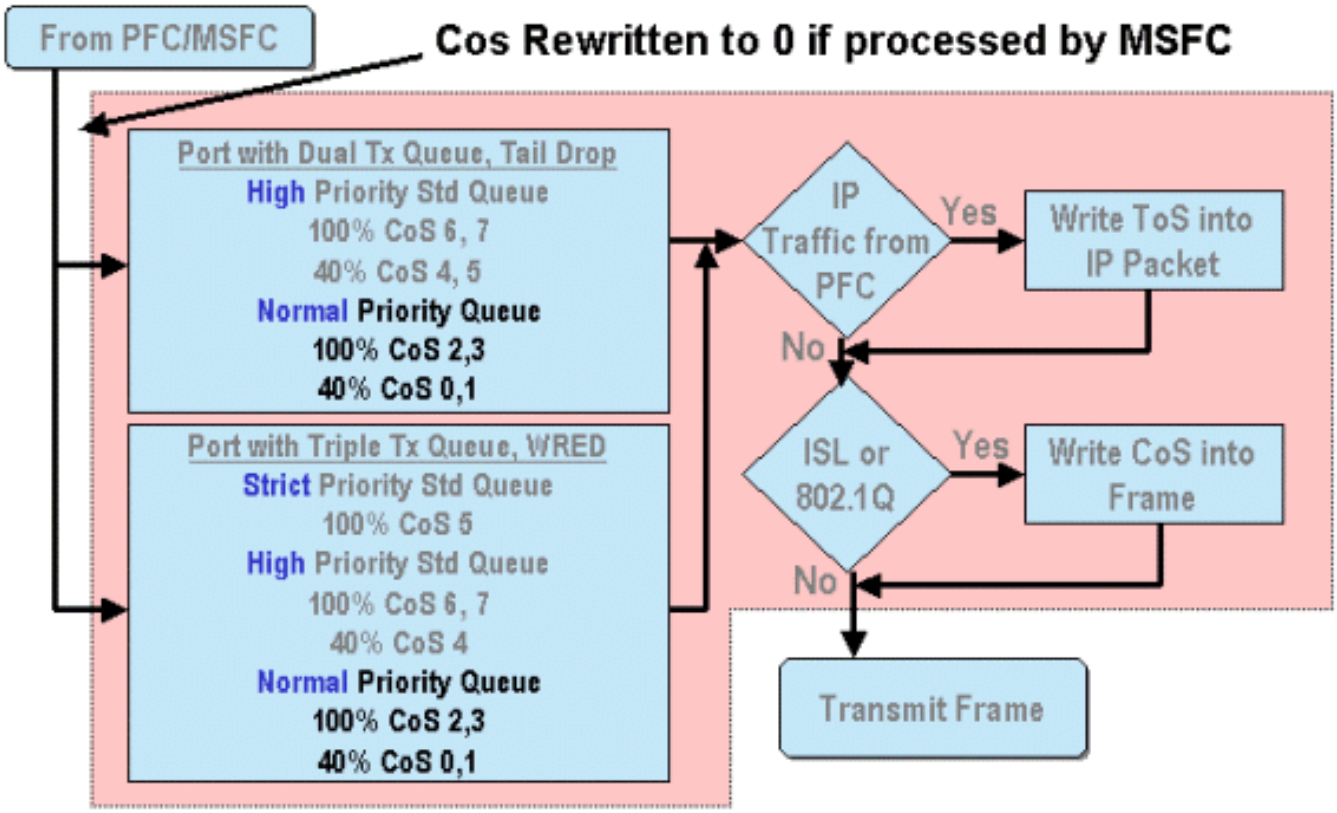
Конфигурация QoS дает команду или ASIC порта или PFC выполнять действие QoS. В следующих разделах будет рассмотрена конфигурация QoS для обоих процессов. На порте ASIC настройка QoS влияет на потоки входящего и исходящего трафика.



Из вышеупомянутой схемы можно заметить, что применяются следующие процессы конфигурации QoS:

1. состояния доверия портов

2. использование CoS на основе порта
3. Назначение порога падения Rx
4 CoS к картам порога отбрасывания Rx



Если кадр обрабатывается MSFC или PFC, он передается на ASIC порта исходящих соединений для дальнейшей обработки. Любым кадрам, обработанным MSFC, перезагрузят их значения CoS для обнуления. Это обстоятельство следует учитывать при обработке QoS на исходящих портах.

Вышеупомянутая схема показывает обработку QoS, выполненную ASIC порта для исходящего трафика. Некоторые процессы, активизированные на исходящей обработке QoS, включают следующее:

1. Назначение отбрасывания остатка TX и порога WRED

2. CoS к отбрасыванию остатка TX и картам WRED

Кроме того, не показанный на схеме выше, процесс переприсвоения CoS к исходящему фрейму с помощью DSCP для Сопоставления CoS.

Следующие разделы исследуют возможности конфигурации QoS основанных ASIC-схем порта более подробно.

Примечание: Важный момент для создания - то, что, когда команды QoS вызваны с помощью CatOS, они, как правило, применяются ко всем портам с указанным типом очереди. Например, если Порог потерь WRED применен к портам с типом очереди 1p2q2 т, этот Порог потерь WRED применен ко всем портам на всех линейных картах, поддерживающих этот тип очереди. В системе Cat IOS команды QoS обычно применяются на уровне интерфейса.

Включение QoS

Прежде чем любая конфигурация QoS может иметь место на Семействе Catalyst 6000, QoS должно сначала быть включено на коммутаторе. Это можно выполнить с помощью следующей команды:

CatOS

```
Console> (enable) set qos enable !-- QoS is enabled. Console> (enable)
```

Интегрированный Cisco IOS (режим работы в собственной системе команд)

```
Cat6500(config)# mls qos
```

Когда QoS будет включено в Семействе Catalyst 6000, коммутатор установит серию настроек по умолчанию QoS для коммутатора. Эти настройки по умолчанию включают следующие параметры настройки:

QoS Feature	Default setting
Trust state of each port	Un-trusted
Receive Queue drop threshold percentages	Threshold 1 – 50% Threshold 2 – 60% Threshold 3 – 80% Threshold 4 – 100%
Transmit Queue drop threshold percentages	Low priority queue threshold 1 – 80% Low priority queue threshold 2 – 100% High priority queue threshold 1 – 80% High priority queue threshold 2 – 100%
CoS value to Drop threshold mapping	Receive queue 1/drop threshold 1: CoS 0 and 1 Transmit queue 1/drop threshold 1: CoS 0 and 1 Receive queue 1/drop threshold 2: CoS 2 and 3 Transmit queue 1/drop threshold 2: CoS 2 and 3 Receive queue 1/drop threshold 3: CoS 4 and 5 Transmit queue 2/drop threshold 1: CoS 4 and 5 Receive queue 1/drop threshold 4: CoS 6 and 7

	Transmit queue 2/drop threshold 2: CoS 6 and 7
CoS to DSCP Mapping (DSCP set from CoS value)	CoS 0 = DSCP 0 CoS 1 = DSCP 8 CoS 2 = DSCP 16 CoS 3 = DSCP 24 CoS 4 = DSCP 32 CoS 5 = DSCP 40 CoS 6 = DSCP 48 CoS 7 = DSCP 56
IP Precedence to DSCP Map (DSCP set from IP Precedence value)	IP precedence 0 = DSCP 0 IP precedence 1 = DSCP 8 IP precedence 2 = DSCP 16 IP precedence 3 = DSCP 24 IP precedence 4 = DSCP 32 IP precedence 5 = DSCP 40 IP precedence 6 = DSCP 48 IP precedence 7 = DSCP 56
DSCP to CoS map (CoS set from DSCP values)	DSCP 0-7 = CoS 0 DSCP 8-15 = CoS 1 DSCP 16-23 = CoS 2 DSCP 24-31 = CoS 3 DSCP 32-39 = CoS 4 DSCP 40-47 = CoS 5 DSCP 48-55 = CoS 6 DSCP 56-63 = CoS 7

Порты, пользующиеся и не пользующиеся доверием

Любой данный порт на Семействе Catalyst 6000 может быть настроен, как доверяется или Недоверяемый, режим доверия порта диктует, как это отмечает, классифицирует и планирует кадр, поскольку это передает транзитом коммутатор. По умолчанию все порты находятся в ненадежном состоянии.

Ненадежные порты (Настройки для портов по умолчанию)

Если порт обозначен в конфигурации как ненадежный, то у кадра, впервые поступающего на порт, ASIC порта обнуляет значения CoS и ToS. Это означает, что кадр будет обрабатываться с низким приоритетом на своем пути через коммутатор.

Также администратор может перезагрузить значение CoS любого Фрейма Ethernet, который вводит ненадежный порт в предопределенное значение. Настройка это будет обсуждено в последующем разделе.

Если порт назначен ненадежным, то коммутатор не будет следить за отсутствием перегрузки. Предотвращение перегрузки является методом, используемым для отбрасывания кадров на основе их значений CoS, как только они превышают пороги, определенные для той очереди. Все кадры, вводящие этот порт, будут одинаково иметь право быть отброшенными, как только буферы достигают 100 процентов.

В CatOS 10/100 или порт GE могут быть настроены как недоверяемые с помощью следующей команды:

CatOS

```
Console> (enable) set port qos 3/16 trust untrusted !-- Port 3/16 qos set to untrusted. Console> (enable)
```

Эта команда переводит порт 16 на модуле 3 в ненадежное состояние.

Примечание: Для Интегрированного Cisco IOS (Режим работы в собственной системе команд) программное обеспечение в настоящее время только поддерживает доверие установки для портов GE.

Интегрированный Cisco IOS (режим работы в собственной системе команд)

```
Cat6500(config)# interface gigabitethernet 1/1 Cat6500(config-if)# no mls qos trust
```

В приведенном выше примере мы вводим конфигурацию интерфейса и не применяем форму команды для установки порта как недоверяемого, так как это - IOS.

Доверенные порты

Время от времени Фреймы Ethernet, вводящие коммутатор, будут иметь или CoS или ToS, устанавливающий, который администратор хочет, чтобы коммутатор поддержал, поскольку кадр передает транзитом коммутатор. Для этого трафика администратор может установить режим доверия порта, где тот трафик входит в коммутатор, как доверяется.

Как отмечалось ранее, коммутатор использует DSCP-значение внутренне для присвоения предопределенного уровня обслуживания на тот кадр. Поскольку кадр вводит надежный порт, администратор может настроить порт для рассмотрения или существующего CoS, приоритета IP-трафика или на DSCP-значения для установки внутреннего значения DSCP. Также администратор может установить предопределенный DSCP в каждый пакет, который вводит порт.

Установить порт в состояние доверяемого можно с помощью следующей команды:

CatOS

```
Console> (enable) set port qos 3/16 trust trust-cos !-- Port 3/16 qos set to trust-COs Console> (enable)
```

Эта команда применима на линейной плате WS-X6548-RJ45. Она устанавливает значение trusted для состояния надежности порта 3/16. Коммутатор будет использовать значение CoS, заданное во входящем кадре, для установки внутреннего DSCP. DSCP получен или из схемы по умолчанию, которая была создана, когда QoS было включено на коммутаторе, или альтернативно из карты, определенной администратором. Вместо ключевого слова trust-cos администратор может также использовать trust dscp или ключевые слова trust-ipprec.

На предыдущих 10/100-линейных картах (WS-X6348-RJ45 и WS-X6248-RJ45) надежность порта должна быть настроена путем выполнения команды set qos acl. В этой команде режим доверия может быть назначен sub параметром **команды set qos acl**. Настройка доверительного CoS на портах этих линейных плат не поддерживается, что показано ниже.

```
Console> (enable) set port qos 4/1 trust trust-COs Trust type trust-COs not supported on this port. !-- Trust-COs not supported, use acl instead. Rx thresholds are enabled on port 4/1. !-- Need to turn on input queue scheduling. Port 4/1 qos set to untrusted. !-- Trust-COs not
```

```
supported, so port is set to untrusted.
```
Команда выше действительно указывает, что требуется, чтобы включать планирование входной очереди. **Итак, для портов 10/100 на линейных картах WS-X6248-RJ45 и WS-X6348-RJ45, все же должна быть настроена команда set port qos x/y trust trust-Cos, хотя для установления состояний доверия нужно использовать ACL.**

С Интегрированным Cisco IOS (Режим работы в собственной системе команд) значение доверия может быть выполнено на интерфейсе GE и 10/100 портах на новой линейной карте WS-X6548-RJ45.

Интегрированный Cisco IOS (режим работы в собственной системе команд)

```
Cat6500(config)# interface gigabitethernet 5/4 Cat6500(config-if)# mls qos trust ip-precedence
Cat6500(config-if)#
```
В этом примере GE-порт 5/4 переводится в доверенное состояние. Для определения значения DSCP будет использоваться значение IP-приоритета кадра.

Классификация входящего трафика и порт установки базирующийся CoS

На входе к порту коммутатора Фрейму Ethernet можно было изменить его CoS, если это соответствует одному из следующих двух критериев:

1. порт настроен как не имеющий доверия или
2. кадр Ethernet не имеет заданного значения CoS

Если вы хотите реконфигурировать CoS входящего кадра Ethernet, необходимо выполнить следующую команду:

CatOS

```
Console> (enable) set port qos 3/16 cos 3 !-- Port 3/16 qos set to 3. Console> (enable)
```
Данная команда устанавливает CO входящих кадров Ethernet в порте 16 в модуле 3 в значение 3, когда прибывает непомеченный кадр или порт устанавливается в состояние "ненадежный".

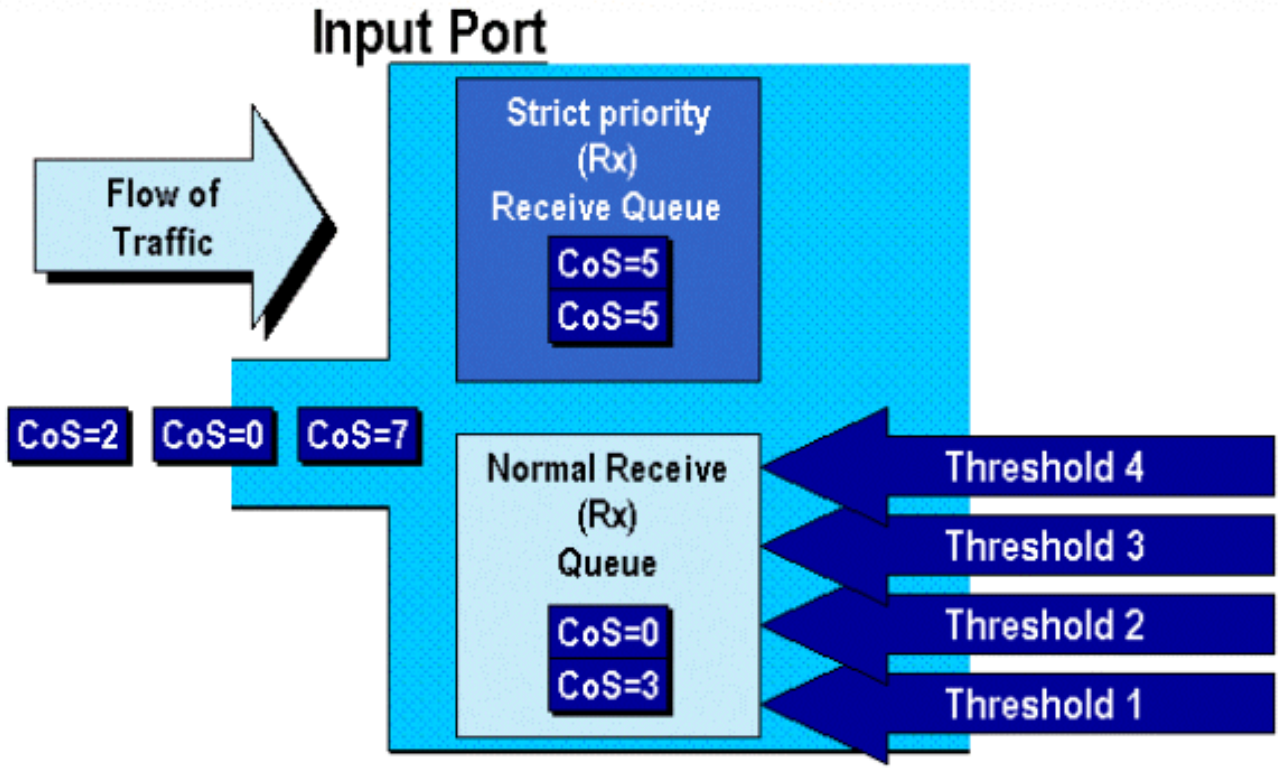
Интегрированный Cisco IOS (режим работы в собственной системе команд)

```
Cat6500(config)# interface fastethernet 5/13 Cat6500(config-if)# mls qos COs 4 Cat6500(config-
if)#
```
Это наборы команд COs входящих кадров Ethernet на порту 13 на модуле 5 к значению 4, когда непомеченный кадр поступает или если порт установлен в недоверяемый.

Настройте пороги отбрасывания Rx

При входе в порт коммутатора фрейм будет помещен в очередь Rx. Чтобы избежать переполнения буфера, на ASIC каждого порта устанавливаются четыре порога для каждой очереди на прием. Эти пороги используются для выявления кадров, которые можно отбросить при их превышении. ASIC порта использует значение CoS кадров, чтобы определить пакеты, которые могут быть отброшены в случае превышения порогового

значения. Эта возможность позволяет кадрам с более высоким приоритетом дольше оставаться в буфере в случае перегрузки.



Как показано в вышеупомянутой схеме, кадры поступают и размещены в очередь. Поскольку очередь начинает заполняться, пороги проверены ASIC порта. При нарушении границы кадры с определенными администратором значениями CO случайным образом отбрасываются из очереди. По умолчанию установлены следующие пороговые значения для очереди 1q4t (на линейных картах WS-X6248-RJ45 и WS-X6348-RJ45):

- порог 1 установлен к 50%, и значения CoS 0 и 1 сопоставлены с этим порогом
- порог 2 установлен к 60%, и значения CoS 2 и 3 сопоставлены с этим порогом
- порог 3 установлен к 80%, и значения CoS 4 и 5 сопоставлены с этим порогом
- для порога 4 установлено значение 100%, а значения 6 и 7 для "CO" согласуются с этим порогом

Для 1P1q4 т (найденный на портах GE) очередь, сопоставления по умолчанию следующие:

- порог 1 установлен к 50%, и значения CoS 0 и 1 сопоставлены с этим порогом
- порог 2 установлен к 60%, и значения CoS 2 и 3 сопоставлены с этим порогом
- порог 3 установлен к 80%, и значения CoS 4 сопоставлены с этим порогом
- для порога 4 установлено значение 100%, а значения 6 и 7 для "CO" согласуются с этим порогом
- Значение CoS 5 сопоставлено с очередью строго по приоритету

Для 1p1q0 т (найденный на 10/100 портах на линейной карте WS-X6548-RJ45), сопоставления по умолчанию следующие:

- Кадры с COs 5 переходят к очереди Rx SP (очередь 2), где коммутатор отбрасывает входящие фреймы только, когда буфер очереди приема SP на 100 процентов полон.
- Кадры с COs 0, 1, 2, 3, 4, 6, или 7 переходят к стандартной очереди Rx. Когда Буфер

очереди rx на 100 процентов полон, коммутатор отбрасывает входящие фреймы.

Данные пороговые значения отбрасывания могут изменяться администратором. Кроме того, значения CoS по умолчанию, которые сопоставлены с каждым порогом, могут также быть изменены. Другие линейные карты внедряют другие реализации очереди Rx. Сводку типов очереди показывают ниже.

CatOS

```
Console> (enable) set qos drop-threshold 1q4t rx queue 1 20 40 75 100 !-- Rx drop thresholds for queue 1 set at 20%, 40%, 75%, and 100%. Console> (enable)
```

Команда задает пределы сбросов при получении для всех входных портов с одной очередью и четырьмя порогами (обозначается 1q4t), равные 20%, 40%, 75% и 100%.

Ниже показана команда, выполняемая в интегрированной Cisco IOS (стандартный режим).

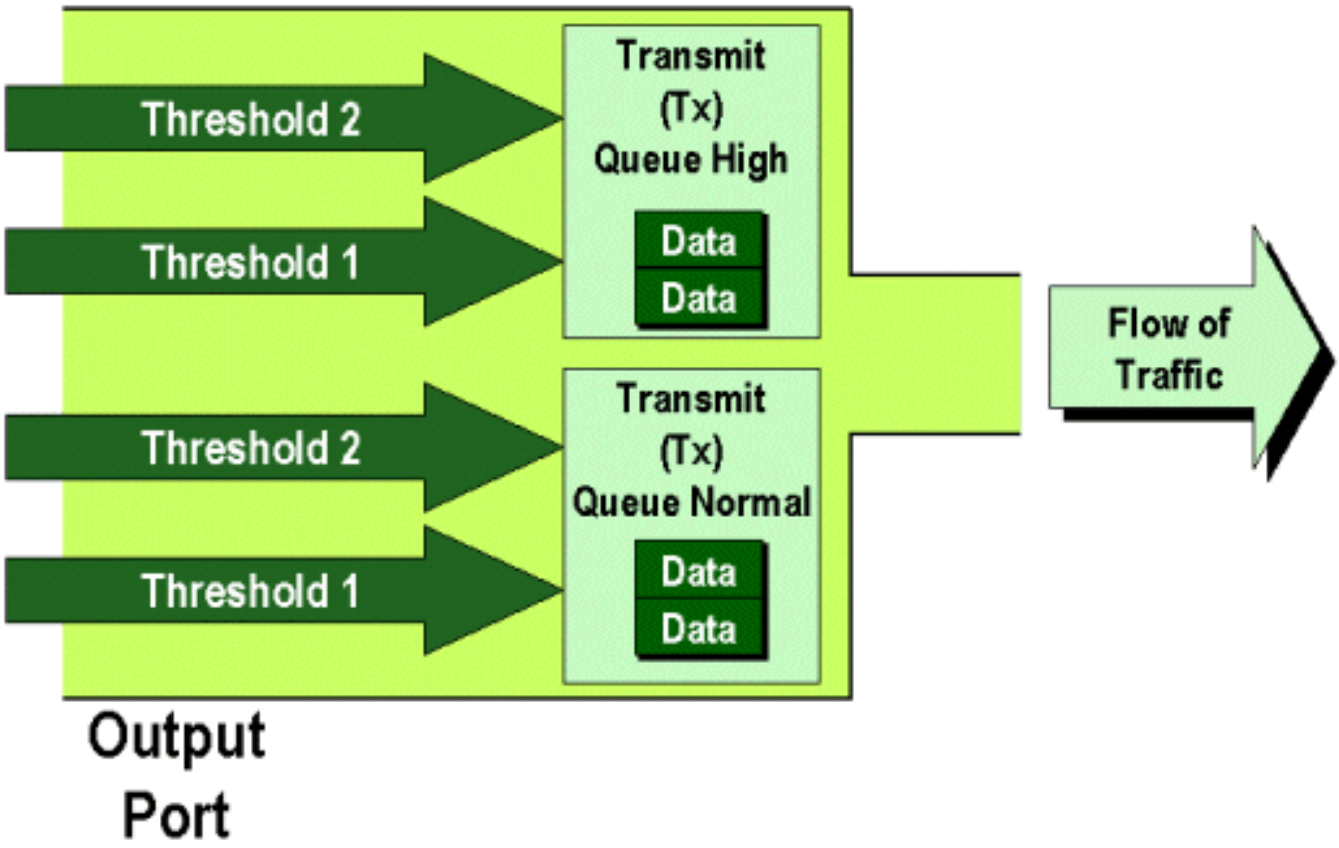
Интегрированный Cisco IOS (режим работы в собственной системе команд)

```
Cat6500(config-if)# wrr-queue threshold 1 40 50 Cat6500(config-if)# wrr-queue threshold 2 60 100 !-- Configures the 4 thresholds for a 1q4t rx queue and. Cat6500(config-if)# rcv-queue threshold 1 60 75 85 100 !-- Configures for a 1p1q4t rx queue, which applies to !-- the new WS-X6548-RJ45 10/100 line card.
```

Пороговые значения отбрасывания Rx должны быть включены администратором. В настоящее время **команда set port qos x/y trust trust-COs** должна использоваться для активации порогов отбрасывания Rx (где *x* является номером модуля, и *y* является портом на том модуле).

Настройка порогов отбрасывания для TX

Для порта выходного трафика предусмотрено два пороговых значения TX, которые используются в механизме предотвращения перегрузок, — очередь 1 и очередь 2. Очередь 1 отвечает стандартной очереди с низким приоритетом, а очередь 2 отвечает стандартной очереди с высоким приоритетом. В зависимости от используемых линейных карт они будут использовать или отбрасывание остатка или алгоритм управления порога WRED. Оба алгоритма используют два порога для каждой очереди TX.



Администратор может вручную установить эти пороги следующим образом:

CatOS

```
Console> (enable) set qos drop-threshold 2q2t TX queue 1 40 100 !-- TX drop thresholds for queue 1 set at 40% and 100%. Console> (enable)
```

Это наборы команд пороги отбрасывания TX для очереди 1 для всех портов вывода с двумя очередями и двумя порогами (обозначает 2q2 т) к 40% и 100%.

```
Console> (enable) set qos wred 1p2q2t TX queue 1 60 100 !-- WRED thresholds for queue 1 set at 60% 100% on all WRED-capable 1p2q2t ports. Console> (enable)
```

Эта команда устанавливает пороги отбрасывания WRED для очереди 1 на всех портах вывода, задавая одну очередь SP, две обычные очереди и два пороговых значения (обозначаются 1p2q2t) на уровне 60% и 100%. Очередь 1 определяется как обычная очередь с низким приоритетом и обладает наименьшим приоритетом. Очередь 2 является высоким приоритетом обычная очередь и имеет более высокий приоритет, чем очередь 1. Очередь 3 является очередью SP и обслуживается перед всеми другими очередями на том порту.

Ниже показана эквивалентная команда, выдаваемая в Integrated Cisco IOS (Native Mode).

Интегрированный Cisco IOS (режим работы в собственной системе команд)

```
Cat6500(config-if)# wrr-queue random-detect max-threshold 1 40 100 Cat6500(config-if)#
```

Это устанавливает пороги отбрасывания WRED для порта 1p2q2t к очереди 1 в 40% для порога 1 (TX) и в 100% для порога 2 (TX).

WRED также можно отключить, если необходимо для интегрированной Cisco IOS

(стандартный режим). Метод использовал делать, это должно использовать **n"** форма команды. Пример отключения WRED показывают следующим образом:

Интегрированный Cisco IOS (режим работы в собственной системе команд)

```
Cat6500(config-if)# no wrr-queue random-detect queue_id
```

Сопоставление MAC-адреса к значениям CoS

В дополнение к установке COs на основе глобального определения порта коммутатор позволяет администратору значениям set COs на основе MAC - адреса назначения и ИДЕНТИФИКАТОРА VLAN. Это обеспечивает кадры, предназначенные для определенных целей, которые будут теговыми с предопределенным значением CoS. Эту конфигурацию можно получить, выдав следующую команду:

CatOS

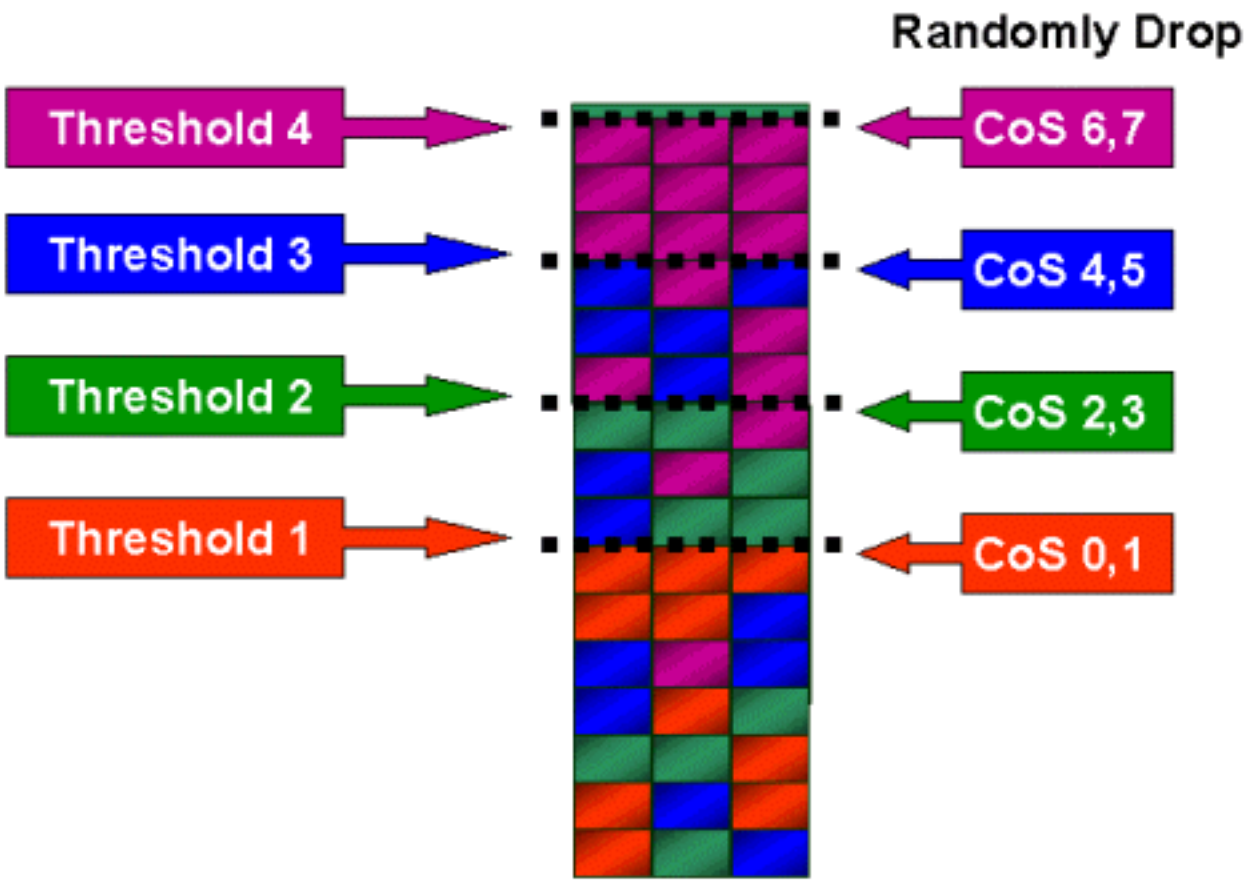
```
Console> (enable) set qos Mac-COs 00-00-0c-33-2a-4e 200 5 !-- COs 5 is assigned to 00-00-0c-33-2a-4e VLAN 200. Console> (enable)
```

Эта команда настраивает 5 CO для любых кадров, MAC адрес назначения которых 00-00-0c-33-2a-4e из виртуальной локальной сети 200.

Нет никакой аналогичной команды в Интегрированном Cisco IOS (Режим работы в собственной системе команд). Так происходит потому, что эта команда поддерживается только в случае отсутствия PFC, а встроенной Cisco IOS (автономный режим) PFC требуется для работы.

Сопоставление COs в пороги

После того, как пороги были настроены, администратор может тогда назначить значения CoS на эти пороги, так, чтобы, когда порог был превышен, могли быть отброшены кадры с определенными значениями CoS. Обычно администратор назначает кадры с более низким приоритетом на более низкие пороги, таким образом, удерживая в очереди высокоприоритетный трафик на случай возникновения затора.



Вышеприведенный рисунок демонстрирует очередь входа с четырьмя порогами и назначение значений CO каждому порогу.

Следующие выходные данные показывают, как значения CO можно преобразовать в пороги:

CatOS

```
Console> (enable) set qos map 2q2t 1 1 COs 0 1 !-- QoS TX priority queue and threshold mapped to COs successfully. Console> (enable)
```

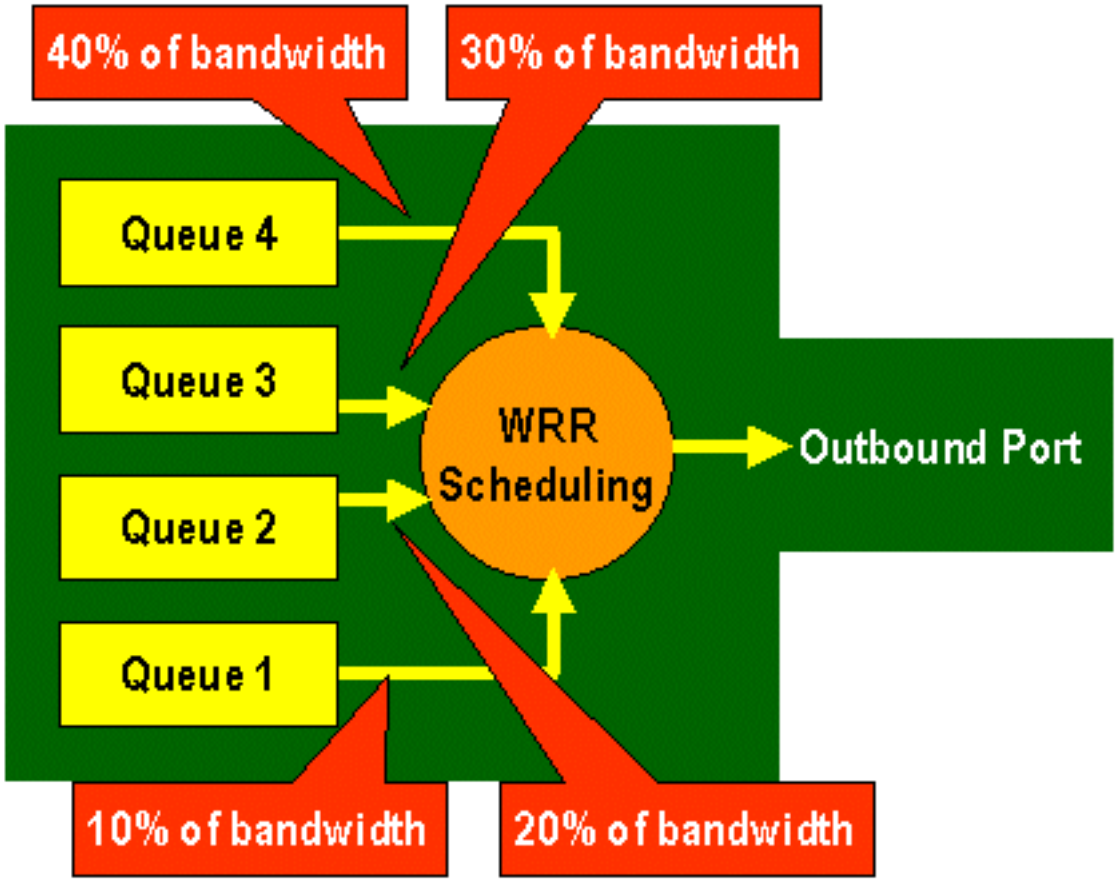
Эта команда назначает значения CoS 0 и 1 помещать в очередь 1, порог 1. Аналогичную команду в Интегрированном Cisco IOS (Режим работы в собственной системе команд) показывают ниже.

Интегрированный Cisco IOS (режим работы в собственной системе команд)

```
Cat6500(config-if)# wrr-queue COs-map 1 1 0 1 Cat6500(config-if)#
```

Настройте пропускную способность в списках очередности TX

Если кадр помещается в исходящую очередь, он будет передан при помощи алгоритма планирования выхода. Процесс планирования вывода использует WRR для передачи кадров из очереди вывода. В зависимости от используемых аппаратных средств линейной карты, существуют или два, три, или четыре очереди передачи на порт.



Механизм WRR на линейных картах WS-X6248 и WS-X6348 (со структурами очереди 2q2t) использует для планирования две очереди TX. На линейных картах WS-X6548 (со структурой очереди на 1p3q1 т) существует четыре очереди TX. Из этих четырех очередей TX три очереди TX обслуживаются алгоритмом WRR (последняя очередь TX является очередью SP). На линейных картах GE существует три очереди TX (использующий структуру очереди на 1p2q2 т); одна из этих очередей является очередью SP так алгоритм WRR только сервисы две очереди TX.

Как правило, администратор назначит вес на очередь TX. WRR работает на основе оценки взвешенности, назначенной очереди порта, которая используется внутренним коммутатором для определения объема трафика, который будет передан до перехода к следующей очереди. Значение надбавки между 1 и 255 может быть назначено на каждую очередь порта.

CatOS

```
Console> (enable) set qos wrr 2q2t 40 80 !-- QoS wrr ratio set successfully. Console> (enable)
```

Эта команда назначает надбавку 40 помещать в очередь 1 и 80 для организации очереди 2. Это эффективно значит два для одного соотношения (от 80 до 40 = от 2 до 1) пропускной способности, назначенной между этими двумя очередями. Данная команда действует на все порты с двумя очередями и двумя порогами на портах.

Ниже показана эквивалентная команда, выдаваемая в Integrated Cisco IOS (Native Mode).

Интегрированный Cisco IOS (режим работы в собственной системе команд)

```
Cat6500(config-if)# wrr-queue bandwidth 1 3 Cat6500(config-if)#
```

Приведенные выше данные соответствуют соотношению 3:1 между двумя очередями. Вы заметите, что версия CAT IOS этой команды применяется к определенному интерфейсу только.

DSCP к сопоставлению COs

Если кадр помещен в выходной порт, порт ASIC будет использовать назначенные CO для избежания перегрузки (т.е. WRED), а также для определения расписания кадра (т.е. передачи кадра). На этом этапе коммутатор будет использовать схему по умолчанию для взятия назначенного DSCP и карты что назад к значению CoS. Эта схема по умолчанию отображена в этой таблице.

Также администратор может создать карту, которая будет использоваться коммутатором, чтобы взять назначенное внутреннее значение DSCP и создать новое значение CoS для кадра. Примеры того, как вы использовали бы CatOS и Интегрированный Cisco IOS (Режим работы в собственной системе команд) для достижения этого, показывают ниже.

CatOS

```
Console> (enable) set qos dscp-cos--map 20-30:5 10-15:3 45-52:7 !-- QoS dscp-cos-map set
successfully. Console> (enable)
```

Вышеупомянутая команда сопоставляет DSCP-значения 20 через к 30 к значению CoS 5, DSCP-значения 10 - 15 к COs 3 и DSCP-значения 45 хотя к 52 к значению CoS 7. Когда QoS было включено на коммутаторе, все другие DSCP-значения используют схему по умолчанию, созданную.

Ниже показана эквивалентная команда, выдаваемая в Integrated Cisco IOS (Native Mode).

Интегрированный Cisco IOS (режим работы в собственной системе команд)

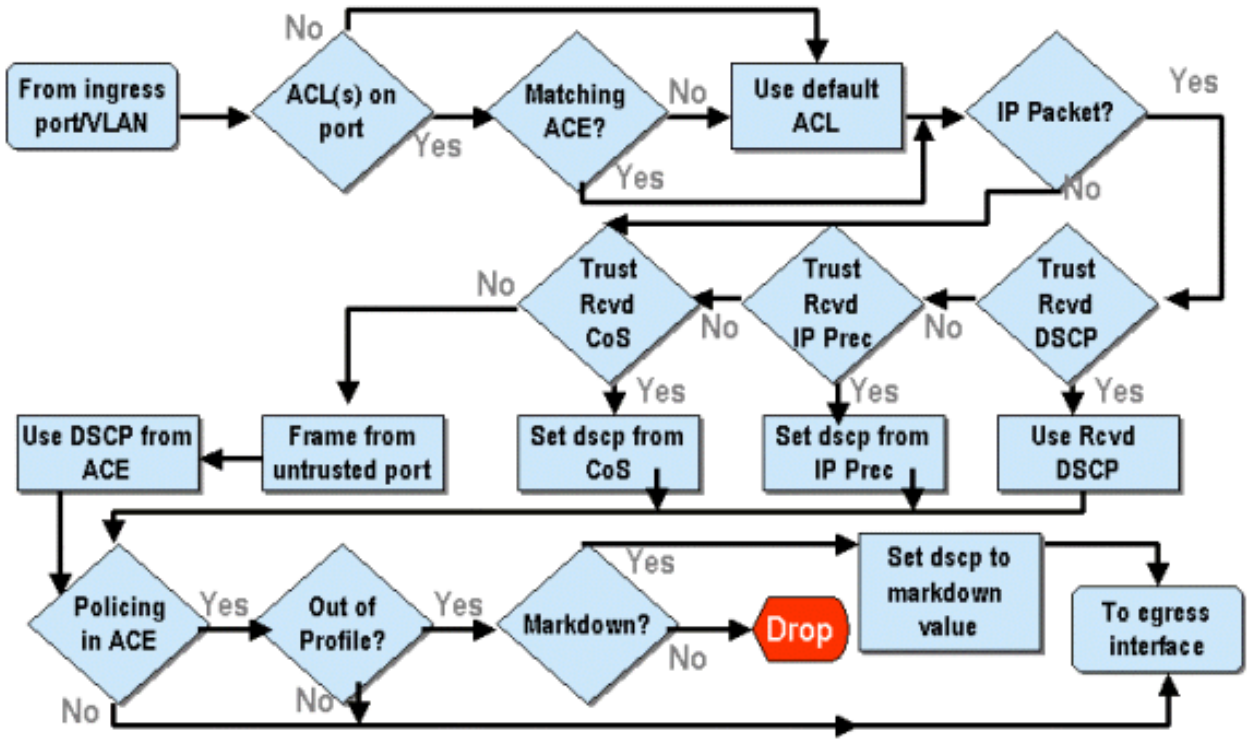
```
Cat6500(config)# mls qos map dscp-cos 20 30 40 50 52 10 1 to 3 Cat6500(config)#
```

Это устанавливает значения DSCP из 20, 30, 40, 50, 52, 10, и 1 для значения COs равного 3.

Классификация и контроль соблюдения правил с помощью PFC

PFC поддерживает классификацию и применение политик кадров. Классификация может использовать ACL для присвоения (отмечают) входящий фрейм приоритетом (DSCP). Применение политик позволяет потоку трафика быть ограниченным определенной величиной пропускной способности.

Следующие разделы опишут эти возможности на PFC и с точки зрения CatOS и с точки зрения Интегрированного Cisco IOS (Режим работы в собственной системе команд) Платформы операционной системы. Процессы, примененные PFC, показывают в следующей схеме:



Настройте применение политик на семействе Catalyst 6000 с CatOS

Функция определения политики разбита на две части: одна для CatOS и одна для интегрированной Cisco IOS (стандартный режим). Оба достигают того же конечного результата, но настроены и внедрены по-разному.

Применение политик

PFC поддерживает способность ограничить (или политика) входящий трафик к коммутатору и может уменьшить поток трафика к предопределенному пределу. Сверхлимитный трафик может быть урезан, или для него может быть понижено значение DSCP в кадре.

Выходные данные (выход) ограничение скорости в настоящее время не поддерживаются или в PFC1 или в PFC2. Это будет добавлено в новой проверке PFC, запланированного на вторую половину 2002, который поддержит выходные данные (или выход) применение политик.

Применение политик поддерживается и в CatOS и в новом Интегрированном Cisco IOS (Режим работы в собственной системе команд), невзирая на то, что конфигурация этих функций очень отличается. В следующих разделах будет описана конфигурация политики обеих платформ OS.

Агрегаты и микропотки (CatOS)

Агрегаты и Микропотки являются терминами, использованными для определения области применения политик, которое выполняет PFC.

Микропоток определяет применение политик единого потока. Поток определен сеансом с уникальным MAC-адресом SA/DA, IP-адресом SA/DA и номерами портов TCP/UDP. Для каждого нового потока, который инициируется через порт VLAN, микропоток может использоваться для ограничения объема данных, полученного для того потока коммутатором. В определении микропотока пакеты, которые превышают заданное предельное значение скорости передачи, могут быть или отброшены или отмечать их

DSCP-значение.

Подобно микропотоку, агрегат данных может использоваться для ограничения скорости трафика. Однако скорость агрегации данных применяется ко всему трафику, входящему на порту или VLAN, которая совпадает с указанным ACL QoS. Можно просмотреть агрегат как применение политик совокупного трафика, который совпадает с профилем в Элементе управления доступом (ACE).

И агрегат и микропоток определяют объем трафика, который может быть принят в коммутатор. И агрегат и микропоток могут быть назначены в то же время на порт или VLAN.

В процессе определения микропотоков можно определить до 63 микропотоков и до 1023 агрегатов.

Записи управления доступом и QoS на базе списков ACL (CatOS)

ACL QoS состоит из списка ACE, определяющих ряд правил QoS что использование PFC для обработки входящих фреймов. Ace подобны Списку контроля доступа маршрутизатора (RACL). ACE определяет критерии классификации, маркировки и применения политик для входящего кадра. Если входящий фрейм совпадет с набором критериев в ACE, то ядро QoS обработает кадр (как считается ACE).

Вся обработка QoS сделана в аппаратных средствах, так включение политик QoS не влияет на производительность коммутатора.

PFC2 в настоящее время поддерживает до 500 ACL, и те ACL могут состоять максимум из 32000 Ace (всего). Фактические первоклассные номера будут зависеть от других определенных сервисов и доступная память в PFC.

Существует три типа Aces, которые могут быть определены. Это IP, IPX и MAC. И IP и Ace IPX осматривают информацию заголовка L3, тогда как MAC базирующиеся Ace только осматривает информацию заголовка L2. Нужно также обратить внимание, что Ace MAC могут только быть применены к не-IP и трафику, отличному от IPX.

Создание правил политики

Процесс создания правила политики влечет за собой создание агрегата (или микропоток), затем сопоставляя тот агрегат (или микропоток) к ACE.

Если, например, требование должно было ограничить весь входящий IP - трафик на порту 5/3 максимум к 20 МБ, два упомянутые выше шага должны быть настроены.

Во-первых, пример запрашивает весь входящий IP - трафик быть ограниченным. Это означает, что необходимо определить общий ограничитель трафика. Пример этого мог бы быть следующие:

```
Console> (enable) set qos policer aggregate test-flow rate 20000 burst 13 policed-dscp !-- Hardware programming in progress !-- QoS policer for aggregate test-flow created successfully.
Console> (enable)
```

Мы создали совокупный поток, назвав его "test-flow". Это определяет скорость 20000 Кбит/с (20 Мбит/с) и пакет 13. Ключевая фраза policed-dscp указывает, что любым данным, превышающим эту политику, отметят ее DSCP-значение, как задано в карте снижения приоритета DSCP (по умолчанию существует, или это может модифицироваться администратором). Альтернатива к использованию ключевой фразы policed-dscp должна

использовать ключевое слово отбрасывания. Ключевое слово drop отбрасывает весь непрофильный трафик (трафик, который выходит за выделенное пиковое значение).

Механизм политик функционирует по схеме "дырявое ведро", для которого определяется всплеск (проходящий объем данных в битах за одну секунду, который будет принят за фиксированный интервал времени), а также скорость передачи (задается как объем данных, выбрасываемый из "ведра" за одну секунду). Любые данные, которые переполняют этого блока, или отброшены или отметили свой DSCP. Упомянутый выше период времени (или интервал) равен 0,00025 секунды (или 1/4000 секунды) и является фиксированным (т. е. это число нельзя изменить, используя команды задания конфигурации).

Номер 13 от приведенного выше примера представляет блок, который примет до 13,000 битов данных каждая 1/4000-я из секунды. Это касается 52 МБ в секунду (13K * (1 / 0.00025) или 13K * 4000). Необходимо всегда следить, чтобы размер пакета был равен или больше скорости передачи исходящих данных. Другими словами, пакет должен быть больше, чем или равным минимальному количеству данных, которые вы хотите передать в течение установленного срока. Если пакет приведет к более низкому рисунку к тому, что вы задали как своя скорость, то ограничение скорости будет равняться пакету. Другими словами, если вы определите скорость 20 Мбит/с и пакет, который вычисляет к 15 Мбит/с, то ваша скорость будет только когда-либо добираться до 15 Мбит/с. Следующий вопрос: почему именно 13? Запомните, пакет определяет глубину области памяти или, другими словами, глубину памяти, используемой для получения входящих данных каждую 1/4000 секунды. Так, пакет мог быть любым номером, поддерживаемым на скорости передачи данных прибытия, больше, чем, или равняться 20 МБ в секунду. Минимальный пакет, который можно использовать для предела скорости 20Мб равен 20000/4000 = 5.

Во время обработки ограничителя алгоритм применения политик сначала заполняет алгоритм token bucket полным набором маркеров. Количество маркеров равно значению пакета. Так, если пиковое значение равняется 13, количество маркеров в блоке равняется 13,000. В течение каждой 1/4000-й из секунды алгоритм применения политик отошлет объем данных, равный определенной скорости, разделенной на 4000. Для каждого бита (двоичный знак) данных передал, это использует один маркер от блока. В конце интервала это пополнит блок новым набором маркеров. Количество маркеров, которые это заменяет, определено скоростью / 4000. Полагайте, что приведенный выше пример понимает это:

```
Console> (enable) set qos policer aggregate test-flow rate 20000 burst 13
```

Предположите, что это - порт на 100 Мбит/с, и мы передаем в постоянном потоке 100 Мбит/с в порт. Мы знаем, что это составит уравнение к скорости передачи входного сигнала 100,000,000 битов в секунду. Параметры здесь являются скоростью 20000 и пакетом 13. Во временном интервале t0, существует полный набор маркеров в блоке (который является 13,000). Во временном интервале t0, у нас будет первый набор данных, поступают в порт. Для на этот раз интервала скорость прибытия будет 100,000,000 / 4000 = 25,000 битов в секунду. Поскольку наш алгоритм Token bucket только имеет глубину 13,000 маркеров, только 13,000 битов 25,000 битов, поступающих в порт в этот интервал, имеют право на то, что были переданы, и 12,000 битов отброшены.

Указанная скорость определяет скорость переадресации 20,000,000 битов в секунду, которая равняется 5,000 битов, передаваемым на 1/4000-й интервал. Для каждых передаваемых 5,000 битов существует 5,000 использованных маркеров. В T1 временного интервала поступают еще 25,000 битов данных, но блок понижается на 12,000 битов. Участок памяти заполняется маркерами, заданными как скорость / 4000 (что эквивалентно 5 000 новым маркерам). Далее алгоритм осуществляет передачу набора данных (другие 5000 бит, для чего требуется еще 5000 маркеров). Процедура повторяется для каждого

интервала.

По существу любые данные, прибывающие сверх глубины ячейки (определенный пакет), отброшены. Данные остались после того, как данные были переданы (соответствие с установленным скорость) также отброшен, освободив дорогу для следующего набора поступающих данных. Неполный пакет является тем, который не был полностью получен во временном интервале, не отброшен, но сохранен, пока это не было полностью получено в порт.

Данный номер пакета предполагает постоянный уровень потока трафика. Однако в реальных глобальных сетях, данные не являются постоянными, и его поток определен размерами окна TCP, которые включают подтверждения TCP в последовательность передачи. Принимая во внимание проблемы с размерами окна TCP, рекомендуется увеличить размер пакета в два раза. В приведенном выше примере предполагаемое значение 13 было бы фактически настроено как 26.

Еще важно отметить, что во временном интервале 0 (то есть в начале цикла применения политики) маркерный бакет заполнен маркерами.

Теперь эта общая политика должна быть интегрирована в a QoS ACE. ACE - то, где спецификация сделана совпасть с рядом критериев к входящему фрейму. Рассмотрим следующий пример. Вы хотите применить описанное выше агрегирование ко всему IP-трафику, но с уточнением по трафику, исходящему из подсети 10.5.x.x и направляющемуся в подсеть 203.100.45.x. ACE будет выглядеть следующим образом:

```
Console> (enable) set qos acl ip test-acl trust-dscp aggregate test-flow tcp 10.5.0.0
203.100.45.0 !-- Test-acl editbuffer modified. Issue the commit command to apply changes.
Console> (enable)
```

Команда выше создала IP ACE (отмеченный использованием команды set qos acl ip command), который теперь связан со списком управления доступом QoS с именем test-acl. Последующие Ace, которые создаются с помощью test-acl и связаны с ним, добавляются в конец списка ACE. К записи в ACE привязан тестовый агрегационный поток. Любым потокам TCP с исходной подсетью 10.5.0.0 и подсетью назначения 203.100.45.0 применятся к этой политике это.

ACL (и связанные Ace) предоставляют очень гранулированный уровень гибкости конфигурации, которую могут использовать администраторы. ACL может состоять из одного или многих Ace, и источник и/или адреса назначения (DA) могут использоваться, а также значения порта L4 для определения отдельных потоков, которые требуются, чтобы охраняться.

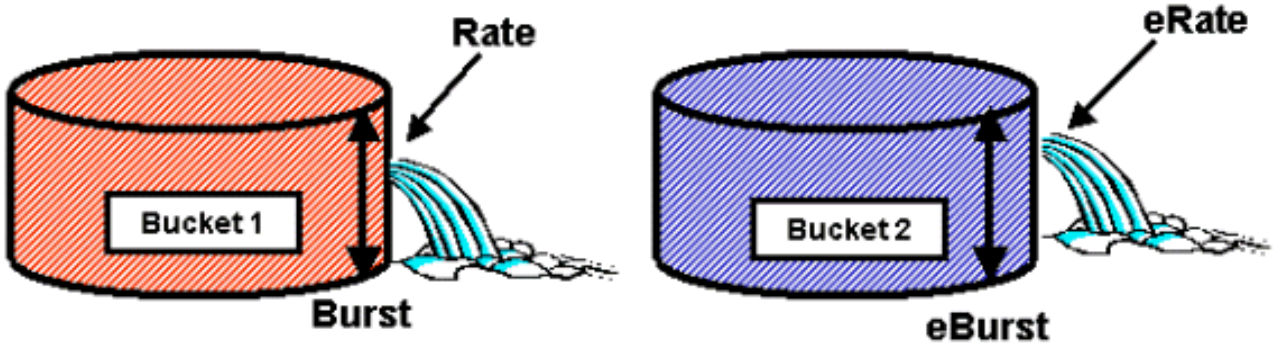
Однако, прежде чем любое применение политик фактически происходит, ACL должен быть сопоставлен или с физическим портом или с VLAN.

Решения о применении политики PFC2

Для PFC2 изменение было внесено в CatOS 7.1 и CatOS 7.2, который представил алгоритм двойного измерения скорости для применения политик. С этим новым алгоритмом это добавляет следующие два новых уровня:

1. **Обычный уровень применения политик:** приравнивается к первому сегменту и определяет параметры, задающие глубину сегмента (блок) и скорость, с которой данные должны передаваться из данного сегмента (скорость).

2. **Избыточный Уровень Применения политик:** это составляет уравнение к второму блоку и определяет параметры, задающие глубину блока (eburst) и скорости, на которой данные должны быть переданы от (сердитого) блока.



Суть процесса заключается в заполнении данными первой области. PFC2 принимает входящий поток данных, меньше чем или равный глубине (пиковое значение) первого блока. Данные, которые переполняются от первого блока, можно отметить и передают к второму блоку. Второй сегмент памяти может принимать скорость входящих данных, поступающих из первого сегмента памяти, в значении, меньшем или равном значению eburst. Данные от второго блока передаются на скорости, определенной erate parameter minus параметр скорости передачи. Данные, которые переполняются от второго блока, могут также быть отмечены или отброшены.

Пример ограничителя двойного измерения скорости следующие:

```
Console> (enable) set qos policer aggregate AGG1 rate 10000 policed-dscp erate 12000 drop burst
13 eburst 13
```

В этом примере используется совокупность AGG1 с избыточной скоростью трафика 10 Мбит/с, которая будет ограничена согласно карте DSCP. Трафик, выходящий за пределы избыточной полосы пропускания erate (установленной на уровне 12 Мбит/с), будет отброшен, поскольку указано ключевое слово drop.

Применение общих ограничителей скорости к модулям с подключенным DFC

Нужно обратить внимание, что приложение общих ограничителей скорости на линейных картах не-DFC может быть достигнуто из-за пути 6000 использования централизованный механизм пересылки (PFC) для перенаправления трафика. Реализация центрального устройства переадресации позволяет сохранять статистику пересылки данных для конкретной VLAN. Этот процесс может использоваться для применения общего ограничителя скорости к VLAN.

На включенной линейной карте DFC, однако, решения по перенаправлению распределены той линейной карте. DFC только знает о портах на своей непосредственной линейной карте и не знает о перемещении трафика на других линейных картах. Поэтому, если общий ограничитель скорости применен к VLAN, которая имеет участвующие порты через несколько модулеи DFC, ограничитель может произвести противоречивые результаты. Причина для этого состоит в том, что DFC может только отслеживать статистику локального порта и не принимает во внимание статистику порта на других линейных картах. Поэтому общий ограничитель скорости применился к VLAN с участвующими портами на включенной линейной карте DFC, приведет к трафику применения политик DFC к номинальному пределу для резидентного объекта портов VLAN на линейной карте DFC только.

Снижение приоритета DSCP сопоставляет (CatOS)

Карты снижения приоритета DSCP используются, когда ограничитель определен к внепрофильному трафику скидки с цены вместо того, чтобы отбросить его. Внепрофильный трафик определен как такой трафик, который превышает установленный параметр пакета сигналов.

Когда QoS включено, карта снижения приоритета DSCP по умолчанию установлена. Карта снижения по умолчанию записана в этой таблице, встречающейся раньше в документе. Интерфейс командной строки (CLI) позволяет администратору модифицировать схему снижения приоритета по умолчанию путем запуска **команды set qos policed-dscp-map**. Пример этой ситуации приведен ниже.

```
Cat6500(config)# set qos policed-dscp-map 20-25:7 33-38:3
```

Данный пример модифицирует охраняемую Карту DSCP, чтобы отразить, что DSCP-значения 20 через к 25 будут отмечены к DSCP-значению 7, и DSCP-значения 33 через к 38 будут отмечены к DSCP-значению 3.

Политика сопоставления к VLAN и портам (CatOS)

После создания список ACL необходимо согласовать с портом или VLAN, чтобы он вступил в действие.

Одна содержательная команда, которая ловит многих не сознающих, является QoS по умолчанию, устанавливающим, который делает весь порт QoS основанным. При применении агрегата (или микропоток) к VLAN он не вступит в силу на порту, пока тот порт не был настроен для основанного QoS VLAN.

```
Console> (enable) set port qos 2/5-10 vlan-based !-- Hardware programming in progress  !-- QoS interface is set to vlan-based for ports 2/5-10. Console> (enable)
```

Изменение QoS на основе портов к QoS на основе VLAN сразу отсоединяет все ACL, назначенные на тот порт, и назначает основанные ACL любой VLAN на тот порт.

Сопоставление ACL к порту (или VLAN) сделано с помощью следующей команды:

```
Console> (enable) set qos acl map test-acl 3/5 !-- Hardware programming in progress  !-- ACL test-acl is attached to port 3/5. Console> (enable) Console> (enable) set qos acl map test-acl 18 !-- Hardware programming in progress  !-- ACL test-acl is attached to VLAN 18. Console> (enable)
```

Даже после сопоставления ACL к порту (или VLAN), ACL все еще не вступает в силу, пока ACL не посвящает себя аппаратным средствам. Это описано в следующем разделе. На этом этапе ACL находится во временном буфере редактирования в памяти. Пока ACL находится в этом буфере, он может быть модифицирован.

Если вы хотите удалить какие-либо незафиксированные ACL, которые находятся в editbuffer, вы выполнили бы **команду отката**. Эта команда, по сути, удаляет ACL из буфера редактирования.

```
Console> (enable) rollback qos acl test-acl !-- Rollback for QoS ACL test-acl is successful.
Console> (enable)
```

Фиксация ACL (CatOS)

Чтобы применить определенный выше QoS ACL, список ACL должен быть зафиксирован на

аппаратном уровне. Процесс фиксации копий ACL с временного буфера на аппаратное обеспечение PFC. После постоянного размещения в памяти PFC политика, определенная в QoS ACL, может быть применена ко любому трафику, который совпадает с записями ACE

Для простоты конфигурации большинство администраторов выполняет **команду commit all**. Однако можно передать определенный ACL (один из многих), который может в настоящее время находиться в буфере редактирования. Пример команды передачи показывают ниже.

```
Console> (enable) commit qos acl test-acl !-- Hardware programming in progress  !-- ACL test-acl is committed to hardware. Console> (enable)
```

Если вы хотите удалить ACL из порта (или VLAN), необходимо очистить карту, которая привязывает тот ACL к тому порту (или VLAN) с помощью следующей команды:

```
Console> (enable) clear qos acl map test-acl 3/5 !-- Hardware programming in progress  !-- ACL test-acl is detached from port 3/5. Console>(enable)
```

Настройка политик на устройствах семейства Catalyst 6000 с встроенной Cisco IOS (собственный режим)

Применение политик поддерживается с Интегрированным Cisco IOS (Режим работы в собственной системе команд). Однако конфигурация и реализация функции контроля достигнуты с помощью карт политик. Каждая карта политик использует классы несколька правил для составления карты политик, и эти классы политики могут быть определены для различных типов трафиков.

Чтобы определить трафик, подлежащий управлению, классы схемы политики в процессе фильтрации используют списки ACL на основе IOS и операторы классового соответствия. После идентификации трафика классы политик могут использовать общие ограничители скорости и ограничители скорости микропотоков для применения политик ограничения скорости к аналогичному трафику.

Следующий раздел содержит подробные сведения о конфигурации политики для интегрированного программного обеспечения Cisco IOS (стандартный режим).

Агрегаты и микропотки (интегрированный Cisco IOS (режим работы в собственной системе команд))

Агрегаты и микропотки являются терминами, использованными для определения области применения политик, которое выполняет PFC. Также как и CatOS агрегаты и микропотоки также используются в интегрированных Cisco IOS (автономный режим).

Микропоток определяет применение политик единого потока. Поток определен сеансом с уникальным MAC-адресом SA/DA, IP-адресом SA/DA и номерами портов TCP/UDP. Для каждого нового потока, который инициируется через порт VLAN, микропоток может использоваться для ограничения объема данных, полученного для того потока коммутатором. В определении микропотока пакеты, которые превышают заданное предельное значение скорости передачи, могут быть или отброшены или отмечать их DSCP-значение. Микропотки применены с помощью команды потока политики, которая является частью класса карты политик.

Чтобы включить управление микропотоком Cisco Integrated IOS (стандартный режим), необходимо включить его глобально на коммутаторе. Этого можно достичь, выполнив следующую команду:

```
Cat6500(config)# mls qos flow-policing
```

Управление микропотоком может также быть применено к проходящему через мост трафик, который является трафиком, который не является коммутированным L3. Чтобы позволить коммутатору поддержать управление микропотоком на проходящем через мост трафик, выполните следующую команду:

```
Cat6500(config)# mls qos bridged
```

Эта команда также включает управление микропотоком для многоадресного трафика. Если многоадресному трафику нужно было примениться к ограничителю скорости микропотоков это, эта команда **(mls qos bridged)** должна быть выполнена.

Подобно микропотоку, агрегат данных может использоваться для ограничения скорости трафика. Однако скорость агрегации данных применяется ко всему трафику, входящему на порту или VLAN, которая совпадает с указанным ACL QoS. Можно рассматривать агрегацию в качестве средства управления совокупным трафиком, который соответствует выбранному профилю трафика.

Есть два вида агрегатов, которые можно определить в интегрированном Cisco IOS (основной режим), как изложено ниже:

- совокупные ограничители скорости для каждого интерфейса
- именованные совокупные ограничители скорости

Агрегаты каждого интерфейса применяются к отдельному интерфейсу с помощью команды police в пределах класса карты политик. Эти классы схем можно применять к нескольким интерфейсам, однако ограничитель скорости контролирует каждый интерфейс по-отдельности. Именованные множества применены к группе портов и определяют политику трафика через все интерфейсы кумулятивно. Именованные множества применены путем запуска **команды mls qos aggregate policer**.

В процессе определения микропотоков можно определить до 63 микропотоков и до 1023 агрегатов.

Создание правил политики (интегрированный Cisco IOS (режим работы в собственной системе команд))

Процесс создания правила политики влечет за собой создание агрегата (или микропоток) через карту политик и затем присоединение той карты политик к интерфейсу.

Считайте тот же пример созданным для CatOS. Требование должно было ограничить весь входящий IP - трафик на порту 5/3 максимум к 20 Мбит/с.

Во-первых, карта политик должна быть создана. Создайте карту политик, названную предельным трафиком. Это сделано следующим образом:

```
Cat6500(config)# policy-map limit-traffic Cat6500(config-pmap)#
```

Вы сразу заметите, что коммутатор побуждает изменения отражать, что вы находитесь в режиме конфигурации для создания класса сопоставления. Помните, что карты политик могут содержать множественные классы. Каждый класс содержит отдельный набор действий политики, которые могут быть применены к другим потокам трафика.

Необходимо создать класс трафика, чтобы специально ограничить входящий трафик до 20 Мбит/сек. Мы вызовем этот класс limit-20. См. пример ниже.

```
Cat6500(config)# policy-map limit-traffic Cat6500(config-pmap)# class limit-to-20
Cat6500(config-pmap-c)#
```

Приглашение снова меняется, чтобы отразить переход к конфигурации класса сопоставления (отображаемой с помощью "-c" в конце приглашения). Если вы хотели применить ограничение скорости для соответствия с определенным входящим трафиком, можно настроить ACL и применить это к имени класса. Если вы хотите применить предел на 20 Мбит/с трафику, полученному от сети 10.10.1.x, выполнить следующий ACL:

```
Cat6500(config)# access-list 101 permit ip 10.10.1.0 0.0.0.255 any
```

Вы могли добавить этот ACL к имени класса следующим образом:

```
Cat6500(config)# policy-map limit-traffic Cat6500(config-pmap)# class limit-to-20 access-group
101 Cat6500(config-pmap-c)#
```

После создания карты классов вы можете приступить к определению индивидуальных политик для этого класса. Можно создать агрегаты (с помощью ключевого слова "police") или микропотоки (с помощью ключевого слова "police flow"). Создайте агрегат, как показано ниже.

```
Cat6500(config)# policy-map limit-traffic Cat6500(config-pmap)# class limit-to-20 access-group
101 Cat6500(config-pmap-c)# police 20000000 13000 confirm-action transmit exceed-action drop
Cat6500(config-pmap-c)# exit Cat6500(config-pmap)# exit Cat6500(config)#
```

Оператор класса, представленный выше (команда police), настраивает ограничения скорости 20000 k (20 Мб/с) с объемом блока данных 52 Мб/с (13000 x 4000 = 52Мб). Если трафик совпадает с профилем и в номинальном пределе, действие должно установить оператором подтверждать-действия для передачи внутрипрофильного трафика. Если трафик внепрофилен (т.е. в нашем приведенном выше примере предел на 20 МБ), оператор exceed-action собирается отбросить трафик (т.е. в нашем примере, весь трафик выше 20 МБ отброшен).

Такое же действие выполняется при настройке микропотока. Если бы мы хотели ограничить все потоки в порт, который совпал с данной картой классов к 200 K каждый, то конфигурация того потока была бы подобна придерживающемуся:

```
Cat6500(config)# mls qos flow-policing Cat6500(config)# policy-map limit-each-flow
Cat6500(config-pmap)# class limit-to-200 Cat6500(config-pmap-c)# police flow 200000 13000
confirm-action transmit exceed-action drop Cat6500(config-pmap-c)# exit Cat6500(config-pmap)#
exit
```

Карты снижения приоритета DSCP

Карты снижения приоритета DSCP используются, когда ограничитель определен к внепрофильному трафику скидки с цены вместо того, чтобы отбросить его. Внепрофильный трафик определен как такой трафик, который превышает установленный параметр пакета сигналов.

Когда QoS включено, карта снижения приоритета DSCP по умолчанию установлена. Карта снижения приоритета по умолчанию приведена в этой таблице. **Интерфейс командной строки (CLI) позволяет администратору изменять разметку по умолчанию с помощью**

команды set qos policed-dscp-map. Пример этой ситуации приведен ниже.

```
Cat6500(config)# mls qos map policed-dscp normal-burst 32 to 16
```

Данный пример определяет модификацию к охраняемой карте DSCP по умолчанию, что DSCP-значение 32 будет отмечено к DSCP-значению 16. Для порта с этим определенным ограничителем любым входящим данные с этим DSCP-значением, которое является частью блока данных сверх установленного пакета, отметят его DSCP-значение к 16.

Политика сопоставления к VLAN и портам (интегрированный Cisco IOS (режим работы в собственной системе команд))

Как только политика была создана, она должна тогда быть сопоставлена или с портом или с VLAN для той политики для вступления в силу. В отличие от процесса передачи в CatOS, нет никакого эквивалента в Интегрированном Cisco IOS (Режим работы в собственной системе команд). После отображения политики на интерфейс эта политика вступает в силу. Для сопоставления вышеупомянутой политики с интерфейсом выполните следующую команду:

```
Cat6500(config)# interface fastethernet 3/5 Cat6500(config-if)# service-policy input limit-traffic
```

Если политика сопоставлена с VLAN для каждого порта в VLAN, которой вы хотите, чтобы Политика виртуальной локальной сети применилась к, необходимо сообщить интерфейсу, что QoS является VLAN, основанной путем запуска **команды mls qos vlan-based**.

```
Cat6500(config)# interface fastethernet 3/5 Cat6500(config-if)# mls qos vlan-based
Cat6500(config-if)# exit Cat6500(config)# interface vlan 100 Cat6500(config-if)# service-policy input limit-traffic
```

Принятие интерфейса 3/5 было частью VLAN 100, политика, названная предельным трафиком, который был применен к VLAN 100, также применится к интерфейсу 3/5.

Настройте классификацию на семействе Catalyst 6000 с CatOS

PFC вводит поддержку классификации данных с помощью списков ACL, которые могут просматривать данные заголовков уровней L2, L3 и L4. Для SupI или IA (без PFC), классификация ограничена использованием трастовых ключевых слов на портах.

В следующем раздела описываются компоненты настройки QoS, используемые PFC для классификации в CatOS.

COs к сопоставлению DSCP (CatOS)

При попадании на коммутатор кадр будет иметь значение DSCP, заданное коммутатором. Если порт будет в надежном состоянии, и администратор использовал ключевое слово trust-cos, то набор значения CoS в кадре будет использоваться для определения набора DSCP-значения для кадра. Как упомянуто прежде, коммутатор может назначить уровни обслуживания на кадр, поскольку это передает транзитом коммутатор на основе внутреннего значения DSCP.

Это ключевое слово на некоторых более ранних 10/100 модулях (WS-X6248 и WS-X6348) не поддерживается. Для тех модулей это рекомендуется с помощью ACL для применения

параметров настройки COs для входящих данные.

При включении QoS коммутатора схема создается по умолчанию. Эта схема используется для определения значения DSCP, которое задается на основе значения центральной ATC. Эти карты перечислены в этой таблице ранее в документе. Также администратор может установить уникальную карту. Пример этой ситуации приведен ниже.

```
Console> (enable) set qos cos-dscp-map 20 30 1 43 63 12 13 8 !-- QoS cos-dscp-map set
successfully. Console> (enable)
```

С помощью приведенной выше команды определяется следующая карта:

COs	0	1	2	3	4	5	6	7
DSCP	20	30	1	43	63	12	13	8

Хотя маловероятно, что в реальной сети указанная выше схема будет использоваться, она позволяет показать, что может быть достигнуто с помощью этой команды.

Приоритет IP-трафика к сопоставлению DSCP (CatOS)

Подобно преобразованию CO в DSCP, кадр может иметь значение DSCP, определенное из параметра приоритета IP входящих пакетов. Это все еще только происходит, если порт установлен в доверяемый администратором, и они использовали ключевое слово trust-ipprec.

При включении QoS коммутатора схема создается по умолчанию. Ссылка на эту карту приведена в данной таблице выше. Эта схема используется для определения значения DSCP, которое задается на основе значения приоритета IP-адреса. Также администратор может установить уникальную карту. Пример этой ситуации приведен ниже:

```
Console> (enable) set qos ipprec-dscp-map 20 30 1 43 63 12 13 8 !-- QoS ipprec-dscp-map set
successfully. Console> (enable)
```

С помощью приведенной выше команды определяется следующая карта:

Приоритет IP	0	1	2	3	4	5	6	7
DSCP	20	30	1	43	63	12	13	8

Хотя маловероятно, что в реальной сети указанная выше схема будет использоваться, она позволяет показать, что может быть достигнуто с помощью этой команды.

Классификация (CatOS)

Когда кадр передается PFC на обработку, процесс классификации выполняется на кадре. PFC будет использовать предварительно настроенный ACL (или ACL по умолчанию) для назначения DSCP фрейму. В ACE одно из четырех ключевых слов используется для присвоения значения DSCP. Список функций:

1. TRUST-DSCP (только IP ACL)
2. TRUST-IPPREC (IP ACL'=s only)
3. TRUST-COS (все ACL, кроме IPX и MAC на PFC2)
4. DSCP

Ключевое слово TRUST DSCP предполагает, что у поступающего в PFC кадра уже есть набор значений DSCP до входа в коммутатор. Коммутатор будет поддерживать это

значение DSCP.

С TRUST-IPPREC PFC получит DSCP-значение из существующего резидентного объекта значения приоритета IP-трафика в поле ToS. Для назначения правильного значения DSCP карта PFC использует IP-приоритет карт DSCP. Когда QoS включено на коммутаторе, схема по умолчанию создана. Также карта, созданная администратором, может использоваться для получения DSCP-значения.

Аналогично TRUST-IPPREC, ключевое слово TRUS-COS инструктирует PFC выводить значение DSCP из CO в заголовке кадра. Также будет доступно преобразование CO в DSCP (по умолчанию или назначенное администратором), чтобы помочь PFC в получении DSCP.

Ключевое слово DSCP используется, когда кадр прибывает с ненадежного порта. Представляет интересную ситуацию для получения DSCP. На этом этапе DSCP, настроенный в операторе set qos acl, используется для получения DSCP. Однако это на этом этапе, где ACL могут использоваться для получения DSCP для трафика на основе набора критериев классификации в ACE. Это значит, что в ACE для определения трафика можно использовать условия классификации, например: IP-адрес источника и получателя, номера портов TCP/UDP, коды ICMP, IGMP-тип, IPX-номера сетей и протоколов, MAC-адреса источников и получателей, а также значения типа работы Ethernet (только для не-IP и не-IPX трафика). Это означает, что ACE мог быть настроен, чтобы назначить определенное DSCP-значение говорить трафик HTTP по трафику FTP.

Рассмотрим следующий пример:

```
Console> (enable) set port qos 3/5 trust untrusted
```

Если порт является ненадежным, PFC используется ACE для извлечения DSCP для кадра. Если ACE настроен с критериями классификации, частное лицо вытекает из того порта, может быть классифицирован с другими приоритетами. Это иллюстрируют следующие ACE:

```
Console> (enable) set qos acl ip abc dscp 32 tcp any any eq http Console> (enable) set qos acl ip ABC dscp 16 tcp any any eq ftp
```

В данном примере у нас есть два оператора ACE. Первое определяет любой поток TCP (ключевое слово, любой используется для определения исходного и конечного трафика), чей номер порта равняется 80 (80 = HTTP), чтобы быть назначенным DSCP-значение 32. Второй ACE определяет трафик, полученный от любого хоста и предназначенный к любому хосту, номер порта TCP которого 21 (FTP) быть назначенным DSCP-значение 16.

Классификация настройки семейства Catalyst 6000 с интегрированным ПО Cisco IOS (основной режим)

Следующий раздел описывает используемые компоненты конфигурации QoS для поддержки классификации на PFC с помощью Интегрированного Cisco IOS (Режим работы в собственной системе команд).

COs к сопоставлению DSCP (интегрированный Cisco IOS (режим работы в собственной системе команд))

При попадании на коммутатор кадр будет иметь значение DSCP, заданное коммутатором. Если порт будет в надежном состоянии, и администратор использовал ключевое слово trust-cos mls qos trust (на портах GE или 10/100 портах на линейных картах WS-X6548), то набор значения CoS в кадре будет использоваться для определения набора DSCP-значения для

кадра. Как упомянуто прежде, коммутатор может назначить уровни обслуживания на кадр, поскольку это передает транзитом коммутатор на основе внутреннего значения DSCP.

При включении QoS коммутатора схема создается по умолчанию. См. в этой таблице параметры по умолчанию. Эта схема используется для определения значения DSCP, которое задается на основе значения центральной ATC. Также администратор может установить уникальную карту. Пример этой ситуации приведен ниже.

```
Cat6500(config)# mls qos map cos-dscp 20 30 1 43 63 12 13 8 Cat6500(config)#
```

С помощью приведенной выше команды определяется следующая карта:

COs	0	1	2	3	4	5	6	7
DSCP	20	30	1	43	63	12	13	8

Хотя маловероятно, что в реальной сети указанная выше схема будет использоваться, она позволяет показать, что может быть достигнуто с помощью этой команды.

Приоритет IP-трафика к сопоставлению DSCP (интегрированный Cisco IOS (режим работы в собственной системе команд))

Подобно преобразованию CO в DSCP, кадр может иметь значение DSCP, определенное из параметра приоритета IP входящих пакетов. Это возможно, только если порт выбран администратором в качестве доверенного и если используется ключевое слово "mls qos trust-ipprec". Это ключевое слово поддерживается только на портах GE и 10/100 на линейных картах WS-X6548. Для 10/100 портов о WS-X6348 и линейные карты WS-X6248, ACL должны использоваться для присвоения ip precedence, доверяют входящим данные.

При включении QoS коммутатора схема создается по умолчанию. См. в этой таблице параметры по умолчанию. Эта схема используется для определения значения DSCP, которое задается на основе значения приоритета IP-адреса. Также администратор может установить уникальную карту. Пример этой ситуации приведен ниже.

```
Cat6500(config)# mls qos map ip-prec-dscp 20 30 1 43 63 12 13 8 Cat6500(config)#
```

С помощью приведенной выше команды определяется следующая карта:

Приоритет IP	0	1	2	3	4	5	6	7
DSCP	20	30	1	43	63	12	13	8

Хотя маловероятно, что в реальной сети указанная выше схема будет использоваться, она позволяет показать, что может быть достигнуто с помощью этой команды.

Классификация (интегрированный Cisco IOS (режим работы в собственной системе команд))

При поступлении кадра на PFC для присвоения ему нового приоритета может быть запущен процесс классификации. В данном случае ограничением является то, что это может быть выполнено, только если кадр получен из ненадежного порта или классифицирован как не заслуживающий доверия.

Действие на основе класса карты политик может использоваться к:

1. Trust cos
2. Trust ip precedence (приоритет доверенных IP-адресов)

3. Trust dscp
4. НИКАКОЕ ДОВЕРИЕ

Ключевое слово TRUST DSCP предполагает, что у поступающего в PFC кадра уже есть набор значений DSCP до входа в коммутатор. Коммутатор будет поддерживать это значение DSCP.

При помощи TRUST IP-PRECEDENCE контролер последовательности команд (PFC) извлечет значение DSCP из существующего значения приоритета IP, находящегося в поле ToS. PFC будет использовать приоритет IP-трафика для Карты DSCP для присвоения корректного DSCP. Когда QoS включено на коммутаторе, схема по умолчанию создана. Также карта, созданная администратором, может использоваться для получения DSCP-значения.

Подобный ТРАСТОВОМУ IP-PRECEDENCE, ключевое слово ДОВЕРИЕ, ПОТОМУ ЧТО дает PFC команду получать DSCP-значение из COs в заголовке фрейма. Также будет доступно преобразование CO в DSCP (по умолчанию или назначенное администратором), чтобы помочь PFC в получении DSCP.

Пример происходящего DSCP от существующего приоритета (DSCP, приоритет IP-трафика или COs) показывают ниже.

```
Cat6500(config)# policy-map assign-dscp-value Cat6500(config-pmap)# class test Cat6500(config-pmap-c)# trust COs Cat6500(config-pmap-c)# exit Cat6500(config-pmap)# exit Cat6500(config)#
```

Приведенная выше схема классов будет получать значение DSCP из поля CoS в заголовке Ethernet.

Когда кадр поступает от ненадежного порта, НИКАКАЯ ТРАСТОВАЯ форма ключевого слова не используется. Это позволяет кадру назначать DSCP-значение во время процесса применения политик.

Рассмотрите следующий пример того, как новый приоритет (DSCP) может быть назначен на другие потоки, входящие в PFC с помощью следующего определения политики.

```
Cat6500(config)# access-list 102 permit tcp any any eq http Cat6500(config)# policy-map new-dscp-for-flow Cat6500(config-pmap)# class test access-group 102 Cat6500(config-pmap-c)# no trust Cat6500(config-pmap-c)# police 1000 1 confirm-action set-dscp-transmit 24 Cat6500(config-pmap-c)# exit Cat6500(config-pmap)# exit Cat6500(config)#
```

Приведенные выше примеры показывают следующее:

1. ACL, создаваемый для определения потоков http, входящих в порт.
2. Карта политик, названная new-dscp-for-flow.
3. Карта классов (называет тест), который использует список доступа 102 для определения трафика, для которого эта карта классов выполнит свое действие.
4. Тест карты класса установит для состояния входящего кадра значение "ненадежное", и присвоит этому потоку DSCP, равную 24.
5. Эта карта классов также ограничит агрегат всех потоков http максимум к 1 МБ.

Стандартный сервер с открытой политикой (COPS)

COPS является протоколом, который позволяет Семейству Catalyst 6000 настроить QoS от удаленного хоста. В настоящее время COPS только поддерживается с помощью CatOS и

является частью архитектуры IntServ для QoS. В настоящий момент (по дате этого документа) в интегрированной Cisco IOS (стандартный режим) нет поддержки COPS. Пока протокол COPS переносит данные конфигурации качества сервиса QoS к коммутатору, он не является источником этих данных. Использование протокола COPS требует, чтобы внешний диспетчер QoS разместил конфигурации QoS для коммутатора. Внешний диспетчер QoS инициирует загрузку этих конфигураций на коммутатор с помощью протокола COPS. QoS Policy Manager (QPM) Cisco является примером внешнего диспетчера QoS.

Это не намерение этого документа объяснить работы QPM, но объяснить конфигурацию, требуемую на коммутаторе поддерживать конфигурации внешнего QoS от использования QPM.

Конфигурация COPS

По умолчанию поддержка COPS отключена. Для использования COPS на коммутаторе это должно быть, включил. Этого можно достичь, выполнив следующую команду:

```
Console> (enable) set qos policy-source cops !-- QoS policy source for the switch set to COPS.
Console> (enable)
```

Если данная команда запущена, определенные значения по умолчанию конфигурации QoS будут запрошены с сервера COPS. Сюда включается следующее:

1. COs к сопоставлениям очередности
2. Назначения пороговых значений очереди ввода и вывода
3. Назначения полосы пропускания WRR
4. Любой агрегат и политика микропотока
5. DSCP к Сопоставлениям CoS для выходного трафика
6. ACL
7. Присвоения порта CoS по умолчанию

Если конфигурации QoS выполняются с помощью COPS, важно помнить, что эти конфигурации применяются по-другому. Вместо того, чтобы настраивать порты напрямую, COPS используется для настройки ASIC порта. Порт ASIC обычно управляет группой портов, поэтому конфигурация COPS применяется одновременно к ряду портов.

Настраиваемый порт ASIC является GE ASIC. На линейных картах GE существует четыре порта на GE (порты 1-4, 5-8, 9-12, 13-16). На этих линейных картах конфигурация COPS влияет на каждую группу портов. На 10/100 линейных картах (как обсуждено ранее в этой газете), существует две группы ASIC-схем, GE и 10/100 ASIC-схем. Один ASIC GE существует для четырех 10/100 ASIC-схем. Каждый 10/100 ASIC поддерживает 12 10/100 портов. COPS настраивает ASIC GE. Таким образом, при применении конфигурации QoS к 10/100 линейным картам через COPS, конфигурация применяется ко всем 48 10/100 портам.

При включении поддержки COPS с помощью команды set qos policy-source cops конфигурация QoS с помощью COPS применяется ко всем ASIC в корпусе коммутатора.
Возможно применить конфигурацию COPS к определенным ASIC-схемам. Это может быть, достигают использования следующей команды:

```
Console> (enable) set port qos 5/4 policy-source cops !-- QoS policy source set to COPS for port
(s) 5/1-4. Console> (enable)
```

Приложение вышеприведенной команды позволяет заключить, что эта команда была

выполнена на модуле GE, поскольку она затронула четыре порта.

Policy Decision Point Servers и доменные имена

Серверы точки принятия решений о применении политики (PDPS) предназначены для внешнего управления политиками и хранения сведений о конфигурации QoS, передаваемые на коммутатор. Если COPS включен на коммутаторе, коммутатор должен быть настроен с IP-адресом внешнего менеджера, который предоставит подробные данные конфигурации QoS к коммутатору. Это схоже с тем, когда SNMP включен и IP-адрес диспетчера SNMP определен.

Команда для определения внешнего PDPS выполняется следующим образом:

```
Console> (enable) set cops server 192.168.1.1 primary !-- 192.168.1.1 is added to the COPS diff-serv server table as primary server. !-- 192.168.1.1 is added to the COPS rsvp server table as primary server. Console> (enable)
```

Приведенная выше команда определяет устройство 192.168.1.1 в качестве сервера точки принятия решений.

Когда коммутатор связывается с PDPS, это должна быть часть домена, определенного на PDPS. PDPS будет только взаимодействовать с коммутаторами, формирующими часть определенного домена так, что коммутатор должен быть конфигурирован для идентификации домена COPS, к которому он относится. Это можно сделать с помощью следующей команды:

```
Console> (enable) set cops domain name remote-cat6k !-- Domain name set to remote-cat6k.
Console> (enable)
```

Приведенная выше команда указывает, что коммутатор настроен как часть домена с именем remote-cat6k. Данный домен должен быть определен в QPM, а коммутатор должен быть добавлен в домен.

Дополнительные сведения

- Поддержка коммутаторов
- Поддержка технологии коммутации локальных сетей
- Cisco Systems – техническая поддержка и документация